Prim'X TECHNOLOGIES

Version 4.0

Cible de Sécurité
Critères Communs niveau EAL3+

Document v1 révision 10

Sommaire

Liste des figures

Liste des tableaux

1. Introduction de la cible de sécurité

1.1. Identification de la cible de sécurité

Cible de sécurité (ST) :	Zed! version 4.0 Cible de sécurité CC niveau EAL3+
Version de la ST :	PX84140 v1r10 – Avril 2010
Cible d'évaluation (TOE) :	- Zed ! v4.0 Edition Standard - Zed ! v4.0 Edition Limitée forme installable - Zed ! v4.0 Edition Limitée forme exécutable pour les plates-formes PC sous Microsoft XP et Vista
Niveau EAL :	**EAL3 augmenté des composants ALC_FLR.3 et AVA_VAN.3 associé à une expertise de l'implémentation de la cryptographie décrite dans [QUALIF_STD].**
Conformité à un PP existant :	Aucune.
Référence des CC :	Critères Communs version 3.1, Parties 1 à 3 – Septembre 2007

1.2. Vue d'ensemble de la cible de sécurité

Zed! est un produit de sécurité permettant de fabriquer des conteneurs de fichiers chiffrés et compressés destinés soit à être archivés soit à être échangés avec des correspondants, en pièces jointes de messages électroniques ou sur des supports variés, comme des clés mémoires USB.

Zed! sera évalué pour une plate-forme PC sous les systèmes d'exploitation Microsoft Windows XP et Vista.

1.3. Conformité aux Critères Communs

Cette cible de sécurité respecte les exigences des Critères Communs version 3.1 de septembre 2007. Tous les composants fonctionnels décrits dans cette cible de sécurité sont issus de la Partie 2 « stricte » des Critères Communs version 3.1 de septembre 2007. Le niveau d'assurance « EAL3 augmenté » retenu est conforme à la Partie 3 « stricte » des Critères Communs version 3.1 de septembre 2007. Le niveau

d'assurance est un niveau EAL3 augmenté des composants ALC_FLR.3 et AVA_VAN.3.

Toutes les interprétations des Critères Communs parues à la date de démarrage de l'évaluation seront retenues.

1.4. Conformité aux référentiels de l'ANSSI

Cette cible de sécurité est conforme aux référentiels de l'ANSSI suivants :

[QUALIF_STD] Processus de qualification d'un produit de sécurité – niveau standard – version 1.2, DCSSI.

[CRYPTO_STD] Mécanismes de cryptographie : règles et recommandations concernant le choix et le dimensionnement des mécanismes cryptographiques - Version 1.20 du 26 janvier 2010, ANSSI.

[CLES_STD] Règles et recommandations concernant la gestion des clés utilisées dans des mécanismes cryptographiques - version 1.10 du 24 octobre 2008, DCSSI

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

2.1. Présentation du produit Zed !

2.1.1. Description Générale

L'email est l'outil le plus utilisé par les entreprises pour communiquer en interne ou avec leurs partenaires et prestataires. Cette communication au quotidien engendre un échange de documents sensibles quasi systématique. Mais, la plupart des documents échangés sont simplement joints aux emails, ne garantissant aucune confidentialité aux données transmises.

Zed! est un **produit de sécurité** pour postes de travail opérant sous Windows XP et Windows Vista. Zed ! se présente comme un produit autonome qui est également intégré dans ZoneCentral. Son rôle est de permettre aux utilisateurs de fabriquer des **conteneurs de fichiers compressés et chiffrés**. Ces conteneurs sont destinés à servir d'archive, ou, plus généralement, de pièce-jointe chiffrée dans des courriers électroniques échangés dans une société.

L'ergonomie est similaire aux fichiers «ZIP» standards sous Windows. L'utilisateur peut déposer des fichiers, les renommer, les supprimer, les extraire, etc. Zed! n'a aucune limite de taille de fichier et ne modifie pas l'arborescence des fichiers ou des dossiers qu'il copie.

Le conteneur permet de gérer un stockage chiffré des fichiers, sans modifier leurs caractéristiques (nom, dates, tailles) et de façon la plus transparente possible pour les utilisateurs. Le chiffrement/déchiffrement des fichiers s'effectue en effet lorsque les fichiers sont lu/copiés dans le conteneur et 'à la volée' (sans manipulation particulière de l'utilisateur).

Après avoir fabriqué un conteneur, l'utilisateur peut ajouter des accès pour ses correspondants. Un accès correspond à une **clé d'accès** (une clé cryptographique) que possède un utilisateur. Cette clé peut être soit un mot de passe "d'échange" convenus avec un correspondant, soit une clé RSA hébergée dans un porte-clés comme un fichier de clé (certificat RSA pris dans un fichier certificat ou recherché sur un annuaire LDAP de certificats), une carte à mémoire, un container CSP Microsoft Windows (le porte-clés pouvant lui-même être protégé par un code confidentiel). Une clé d'accès permet de retrouver (en les déchiffrant) les informations de chiffrement des fichiers du conteneur.

Zed! se décline en différents packages :

- L'Edition Standard, qui contient le produit complet ;
- L'Edition Limitée (appelé «**Zed! Edition Limitée**»), gratuite, libre de distribution et d'usage, qui permet aux correspondants de lire le contenu des conteneurs (moyennant la fourniture d'une clé d'accès) et d'en extraire les fichiers. Le correspondant a également le droit de modifier le contenu du

conteneur (enlever, ajouter des fichiers) pour pouvoir le renvoyer à l'émetteur d'origine. L'édition limitée ne lui permet pas, cependant, de créer de nouveaux conteneurs ou d'en modifier les accès prévus par le créateur original du conteneur. L'Edition Limitée existe sous deux formes :

  - Une forme installable (i.e. avec programme d'installation) et intégrée à l'Explorateur Windows ;
  - Un simple exécutable, facile à transporter, et qui évite d'avoir à effectuer une installation ;

- Enfin, Zed! est également incorporé dans les différents produits de la gamme ZoneCentral.

2.1.2. La technologie de Zed!

Un conteneur chiffré Zed! possède une logistique plus légère que le produit ZoneCentral (développé par la même société) tout en en reprenant certains éléments. Par exemple, une des différences fondamentales est que le cryptosystème utilisé est 'embarqué' en mode User (et non pas en mode 'Kernel') et qu'il n'y a pas de 'mémorisation de contexte de clé d'accès' (deux ouvertures du même conteneur vont entrainer deux fournitures de clés d'accès alors que ZoneCentral sait réutiliser une clé d'accès déjà fournie).

Le format interne du conteneur permet de contenir des fichiers de toutes tailles, indépendamment les uns des autres, et il gère en interne le nommage de ces fichiers. Le conteneur peut être considéré comme un "dossier virtuel" : son contenu en termes de fichiers n'est pas confidentiel, c'est le contenu des fichiers qui l'est. Le conteneur gère en interne un fichier de contrôle qui est masqué.

2.1.3. Les conteneurs et les accès

Chaque conteneur est définie par certaines caractéristiques de chiffrement (dont font partie les clés de chiffrement des fichiers, les algorithmes, etc.) et une liste d'accès utilisateurs. La définition des accès est libre, mais le produit est doté de fonctions et de mécanismes d'administration permettant d'imposer certains accès ou certains types d'accès.

Pour pouvoir utiliser un conteneur, un utilisateur doit donc disposer d'une clé d'accès ou d'un mot de passe. Dans le cas de Zed ! Edition Standard, cette clé d'accès lui a été remise par l'Administrateur de la Sécurité (appelé Administrateur de la TOE dans la suite du document). Il peut s'agir d'une clé RSA hébergée dans un porte-clés comme un fichier de clés, une carte à puce, un container Microsoft CSP (le porte-clés intégrant la plupart du temps son propre dispositif d'authentification avec un code confidentiel). Le mot de passe est le plus souvent choisi par l'utilisateur en fonction de la politique de sécurité mise en œuvre.

Concernant Zed ! Edition Limitée, c'est plutôt l'utilisateur émetteur qui gère les droits du destinataire.

Une fois l'utilisateur authentifié, la clé d'accès ainsi fournie reste valide tant qu'elle n'a pas été explicitement fermée par l'utilisateur (fermeture de l'explorateur, arrêt du système par exemple).

Lorsque le conteneur a été fabriqué, les fichiers protégés par le conteneur ont été chiffrés avec des clés elles mêmes chiffrées avec les clés d'accès des utilisateurs. Bien entendu, les clés d'accès elles-mêmes ne figurent pas dans le conteneur.

Zed ! propose différents algorithmes et mécanismes de sécurité, tous conformes à l'état de l'art en la matière. Il propose deux schémas de gestion de clés d'accès qui peuvent être utilisés en même temps. Un schéma dit « symétrique » basé sur des mots de passe et des clés dérivées de mots de passe (réf. : PKCS#5) et un schéma dit « asymétrique » utilisant des clés RSA (réf. : PKCS#1 v1.5) embarquées dans des fichiers de clés (réf. : PKCS#12) ou des porte-clés (ref: PKCS#11 et/ou CSP).

2.2. Services d'utilisation et rôles

2.2.1. Définition des rôles

Hormis le responsable de la sécurité de l'organisation qui fixe la politique générale de sécurité à appliquer, on distingue 3 rôles mettant en œuvre (directement ou indirectement) les fonctionnalités de la TOE :

- Un rôle opérant uniquement dans l'environnement de la TOE (ce rôle ne concerne pas Zed ! Edition Limitée) : L'administrateur de la sécurité de l'environnement Windows des utilisateurs (administrateur Windows) en charge de définir les règles d'usage et de sécurité (les policies), c'est-à-dire le paramétrage de fonctionnement du produit : cette opération de « haut-niveau » est effectuée sous le contrôle du Responsable de la Sécurité qui a étudié les différents paramètres et décidé des valeurs à affecter pour obtenir le comportement souhaité du produit dans le cadre d'utilisation et d'environnement prévu. Ces règles ne changeront ensuite que de façon très exceptionnelle. Il est à noter que ce rôle peut se décliner en plusieurs rôles hiérarchiques correspondant aux différents niveaux des domaines Windows. Dans ce cas les administrateurs des niveaux supérieurs doivent interdire aux administrateurs des sous-niveaux (domaines, contrôleurs de domaines, postes de travail) la modification des « policies » de la TOE qu'ils souhaitent eux-mêmes contrôler.
- Un rôle administrateur de la TOE (ce rôle ne concerne pas Zed ! Edition Limitée) en charge de l'installation de la TOE, des opérations de recouvrement et de la mise à disposition des clés d'accès et éventuellement des mots de passe. Sauf mention contraire dans la suite de ce document, toute référence à l'administrateur se rapporte à ce rôle.
- Un rôle utilisateur qui utilise la TOE selon la configuration imposée.

Il faut noter que, à part la définition des policies, généralement dévolue à un responsable de la sécurité, les autres opérations peuvent être effectuées par différents acteurs en fonction de la confiance, de l'organisation et des moyens de l'entreprise.

2.2.2. Services d'utilisation d'un conteneur

Il n'y a pas d'opération d'administration sur un conteneur autres que les gestions d'accès. Un conteneur "vit" tant qu'il existe (fichier conteneur non supprimé).

Les opérations possibles sont :

- La création d'un nouveau conteneur (par l'utilisateur). Cette opération ne requiert aucune clé d'accès. Par ailleurs la fonction n'est pas mise à disposition dans Zed ! Edition Limitée.
- L'initialisation du conteneur à la première tentative d'accès (correspond à la création du premier accès).
- L'ajout d'un accès utilisateur (ce qui implique que sa clé d'accès soit fournie). Cette opération ne concerne pas Zed ! Edition Limitée.
- Le renommage ou la suppression d'un conteneur.
- L'ajout de fichiers au conteneur (drag&drop, insérer, copier/coller, etc.) : les fichiers sont chiffrés pour le conteneur quand ils sont mis dans le conteneur, ce qui nécessite d'avoir fourni la clé d'accès du conteneur.
- La visualisation ou l'extraction de fichiers d'un conteneur : toute "sortie" de fichier nécessite la clé d'accès du conteneur pour que les fichiers soient déchiffrés du conteneur.
- La suppression de fichiers du conteneur.
- La création et la suppression de dossiers dans le conteneur

L'interface d'utilisation des conteneurs chiffrés est tout à fait similaire à l'utilisation des 'dossiers compressés' (.zip) sous Windows XP (glisser-déplacer, copier-coller …).

2.2.3. Exemple d'utilisation de Zed ! Edition Standard

Il existe différents scénarios de mise en œuvre, mais le principe d'utilisation reste le même pour les utilisateurs.

L'administrateur Windows définit ***les règles d'usage (policies)*** du produit, ce qui se traduit par une configuration prédéfinie (policy) qui peut être masterisée (personnalisation de l'installation) ou télé-gérée (diffusée, mise à jour) soit par des commandes d'administration fournies par le produit soit par la logistique intégrée des réseaux bureautiques (exemple : contrôleurs de domaines). Ces règles sont

généralement établies à « haut niveau » dans l'entreprise par le Responsable de la Sécurité. Parmi ces règles, on trouve, par exemple, l'algorithme de chiffrement à utiliser, le comportement que doit adopter le logiciel dans certains cas, les porte-clés PKCS#11 supportés etc.

Le logiciel, masterisé ou non, est ensuite ***installé*** sur un poste de travail, manuellement ou via les logiciels de télé-installation du marché.

Par ailleurs, il est à la charge de l'administrateur de la TOE de ***définir (fournir) les clés d'accès*** des utilisateurs (issues d'une PKI, par exemple). Zed! supporte différents scénarios de gestion de clés, mais n'en fournit pas l'infrastructure. Si une PKI est en place, il sait en utiliser les éléments (clés RSA, porte-clés, certificats), si elle n'est que partiellement installée, ou s'il n'y en a pas, il sait également utiliser des accès par mots de passe.

Seuls les utilisateurs disposant de clés d'accès valides pour le conteneur chiffré pourront lire ou écrire des fichiers dans celui ci. A la première tentative d'accès à un fichier chiffré dans le conteneur, Zed! demande à l'utilisateur une clé d'accès permettant de déchiffrer le fichier (en pratique, le schéma est plus complexe, et cette clé d'accès permet de déchiffrer des clés intermédiaires qui elles-mêmes chiffrent les fichiers). Si l'utilisateur peut la fournir, alors le fichier peut être déchiffré (ou chiffré, s'il s'agit d'une création ou d'une écriture). Sinon, l'utilisateur se voit refuser l'accès avec le message « Accès non autorisé». L'authentification de l'utilisateur auprès du conteneur reste effective jusqu'à la fermeture de celui-ci par fermeture de l'explorateur, clôture de la session Windows ou arrêt du poste. Comme pour les archives zip, pour « ouvrir » (lire) un fichier d'un conteneur, il doit d'abord être extrait. L'opération « ouvrir » automatise l'extraction (dans le dossier temp de l'utilisateur) et l'active. Quand le conteneur est refermé, le fichier est chiffré.

2.3. Périmètre et architecture de la cible d'évaluation

2.3.1. Les composants de Zed !

La figure 1 présente l'architecture du produit : le périmètre de la TOE est délimité par des pointillés.

«**Interface Explorer**» implémente les interfaces Shell de Windows qui permettent de gérer les menus et la vue graphique accessibles à partir de l'explorateur Windows.

«**ZCC**» est le centre cryptographique de Zed! : il gère les clés et exécute les opérations de calcul associées. Les clés ne sortent jamais de son enceinte, sauf lorsque le produit est configuré pour utiliser des porte-clés (comme des extensions PKCS#11 pour des cartes à puce ou des CSPs).

«**ZCCA**» référence les clés utilisateur saisies via l'entrée d'un mot de passe, l'interface PKCS#11 ou le CSP.

«**Moteur Zed**», coordonne les différents traitements;

«**Langues**» représente les dlls associées aux différentes langues supportées par le produit.

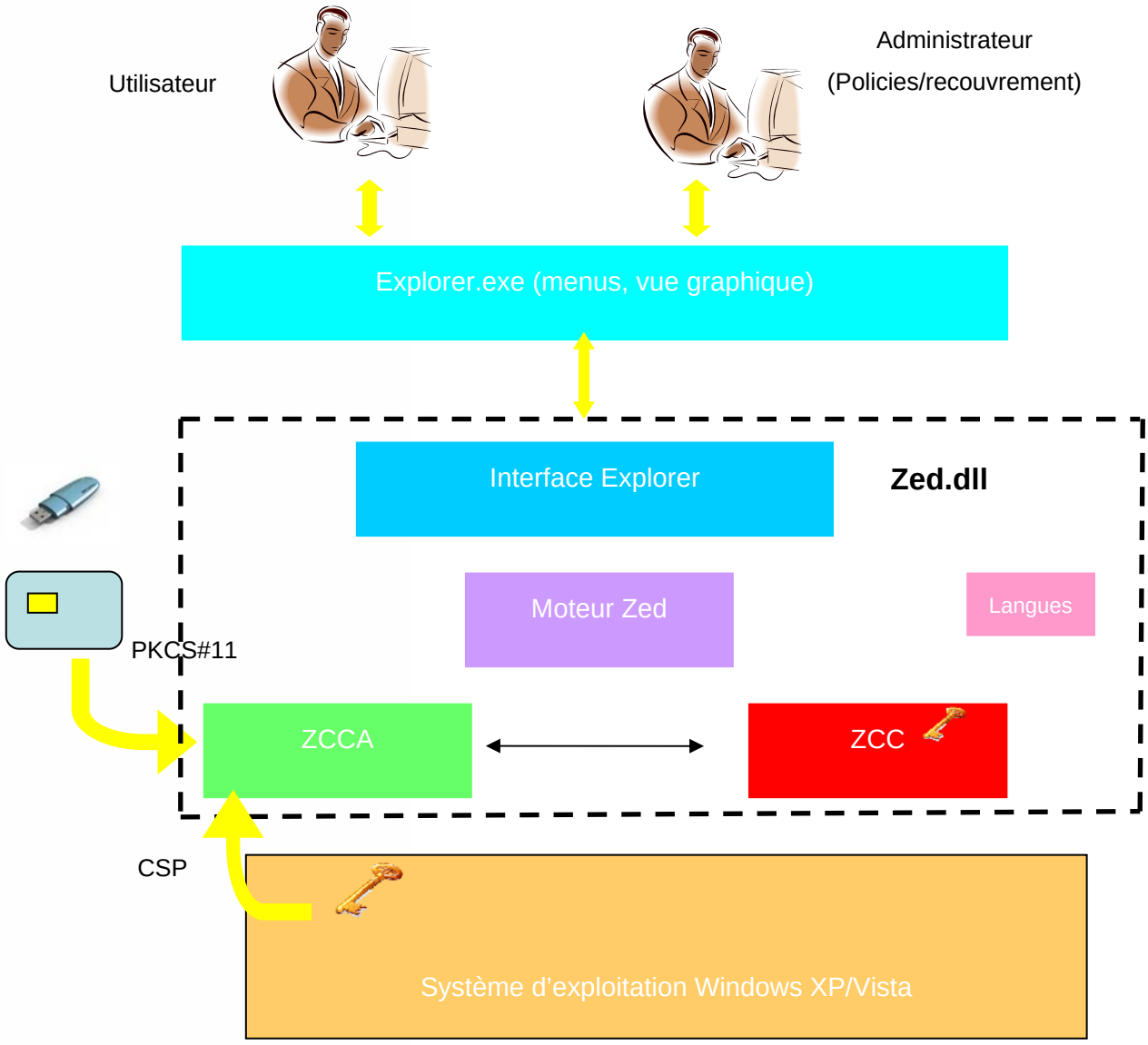

Figure 1 – Schéma de principe de la TOE

2.3.1.1 Périmètre logique

La TOE est constitué de Zed! Edition Standard et Zed! Edition Limitée en version installable et exécutable (voir chapitre 2.1.1). Le produit intégré dans ZoneCentral ne fait donc pas partie du périmètre de l'évaluation.

Le périmètre d'évaluation est constitué de l'ensemble des composants du logiciel Zed!

2.3.1.2 Périmètre physique

Zed ! sera évalué, en tant que produit, sur une plate-forme PC sous les systèmes d'exploitation de Microsoft suivants : Windows XP et Windows Vista.

L'utilisation avec les différentes clés d'accès sera évalué (mot de passe et clé RSA). En particulier, le dialogue PKCS#11 entre la TOE et les porte-clés utilisateurs, le dialogue PKCS#12 entre la TOE et les fichiers de clés, le dialogue réseaux entre la TOE et les données utilisateurs stockées sur des médias distants (serveur sur un réseau local ou sur Internet par exemple) seront également évalués.

Les éléments suivants sont hors évaluation :

- Le dialogue clavier entre la TOE et la saisie des mots de passe ;
- Les systèmes d'exploitation Windows, y compris :
  - Les drivers PC/SC ;
  - Le service de gestion des certificats (CMS) ;
  - Le service de gestion des profils utilisateurs (User management) ;
- Les portes clés utilisés (comme les porte-clés de type Token USB, les fichiers de clés ou les containers CSP).

Le logiciel Zed! utilise des clés utilisateurs (les «clés d'accès») qui peuvent être fournies par l'environnement (clés RSA dans des porte-clés utilisateur).

2.4. Plate-forme de tests pour l'évaluation de la TOE

Pour l'évaluation du produit Zed!, la plate-forme minimale suivante devra être mise en place par l'évaluateur. Le type physique de porte-clés (carte à puce ou clé USB) étant transparent pour Zed! (seul le dialogue PKCS#11 est important), les tests de l'évaluateur pourront s'effectuer avec un seul type de porte-clés.

PC1 configuré avec Zed!

PC2 configuré avec Zed! et/ou Zed! Edition Limitée

Figure 2 – Plate-forme de tests pour l'évaluation de la TOE

3. Définition du problème de sécurité

3.1. Les biens sensibles

3.1.1. Biens sensibles de l'utilisateur

3.1.1.1 Clés d'accès

Pour ouvrir (lecture, remplissage, gestion des accès) les conteneurs chiffrés, Zed! met en œuvre les clés d'accès des utilisateurs. En fonction des cas de figure, il peut être amené à manipuler directement soit le mot de passe utilisateur, soit la clé d'accès elle-même, soit le code confidentiel de protection de la clé d'accès.

- Accès par mot de passe : Zed! gère la saisie du mot de passe, sa transformation (dérivation) en clé d'accès puis le déchiffrement de la clé de chiffrement et déchiffrement des fichiers du conteneur par cette clé d'accès ;
- Accès par clé RSA hébergée dans un fichier de clés : Zed ! gère la saisie du code confidentiel du fichier de clés, lit et déchiffre le fichier de clés avec ce code confidentiel, obtient la clé d'accès RSA et effectue le déchiffrement de la clé de chiffrement et déchiffrement des fichiers du conteneur par cette clé ;
- Accès par clé RSA hébergée dans un token logique accédé au travers d'un composant externe PKCS#11 (ce composant pouvant piloter une carte à mémoire, un token USB ou tout autre dispositif hardware ou software) : Zed! gère la saisie du code confidentiel du token logique, le remet au composant externe pour le déverrouiller, il n'accède pas à la clé RSA et n'effectue pas lui-même le déchiffrement de la clé de chiffrement et déchiffrement des fichiers avec cette clé, celui-ci est effectué par le composant externe qui communique ensuite la clé obtenue à Zed !;
- Accès par clé RSA hébergée dans un token logique accédé au travers d'un composant externe CSP (ce composant pouvant piloter une carte à mémoire, un token USB ou tout autre dispositif hardware ou software) : Zed! ne gère pas la saisie du code confidentiel du token logique, c'est le composant externe qui le fait spontanément avec ses propres moyens, et il n'accède pas à la clé RSA et n'effectue pas lui-même le déchiffrement de la clé de chiffrement et déchiffrement des fichiers avec cette clé, celui-ci est effectué par le composant externe qui communique ensuite la clé obtenue à Zed !;

En fonction de ces cas, donc, Zed! manipule comme biens sensibles un mot de passe ou code confidentiel (en saisie), et une clé d'accès cryptographique. Dans les cas 1 et 2, il manipule les deux éléments, dans le cas 3, il ne manipule que le premier, dans le cas 4, il n'en manipule aucun.

Il faut noter que Zed! ne génère PAS les clés d'accès des utilisateurs : quand il s'agit de clés RSA, quel que soit le porte-clés qui les héberge et le module qui les traite, elles sont toujours générées par un outil externe à Zed! (en général une PKI), de même que le porte-clés éventuel et le code confidentiel de protection. Quand il s'agit de mots de passe, c'est le plus souvent l'utilisateur qui le choisit. L'utilisateur et son environnement (règles et procédures internes, établies par le Responsable de la

Sécurité) sont responsables de la qualité de ces clés, de la protection du porte-clés et de leur bonne utilisation.

3.1.1.2 Fichiers chiffrés

Zed! permet de conserver sous forme chiffrée les fichiers et dossiers. Les biens sensibles sont donc les fichiers et dossiers utilisateurs, de tous types, stockés dans les conteneurs.

Les fichiers ainsi chiffrés dans les conteneurs sont des biens sensibles de l'utilisateur protégés par la TOE (qui doit conserver leur image stockée chiffrée sans copie en clair) tant qu'ils demeurent dans leur conteneur.

3.1.2. Biens sensibles de la TOE

3.1.2.1 Les programmes

Pour assurer son fonctionnement, la TOE met en œuvre ses ***programmes*** (exécutables, bibliothèques dynamiques). La sécurité en intégrité de ces programmes est assurée par l'environnement : il faut être administrateur Windows pour les modifier. Ces programmes sont également signés (système authenticode Windows).

3.1.2.2 La configuration

Pour assurer son fonctionnement, la TOE met en œuvre des policies (au sens Windows du terme). La sécurité en intégrité de ces policies est assurée par l'environnement (i.e. le système des policies sous Windows) : il faut être l'administrateur Windows de plus haut niveau pour les modifier (si un domaine Windows définit une valeur pour un paramètre, alors un administrateur local au poste ne pourra pas la modifier).

3.1.2.3 Les fichiers de contrôle

Il s'agit de fichiers décrivant les conteneurs chiffrés. Ces fichiers contiennent le libellé du conteneur, un identifiant unique, quelques informations de gestion, et les 'wrappings' d'accès, c'est-à-dire les clés de chiffrement du conteneur chiffrées par les clés d'accès des utilisateurs habilités.

3.1.3. Synthèse des biens sensibles

Le tableau ci-dessous résume la liste des biens sensibles protégés par Zed! et indique la nature de la sensibilité associée.

Remarque : de façon générale, l'intégrité n'est pas un objectif de Zed!. Le rôle du produit est de gérer la confidentialité des biens sensibles qui lui sont confiés, mais ce n'est pas un produit dont le but est de détecter une altération quelconque dans l'environnement (intrusion, virus, etc.). Par contre, Zed ! met en œuvre des dispositifs permettant de détecter des altérations qui seraient nuisibles à son bon fonctionnement, ou qui induiraient un défaut dans son objectif de confidentialité.

Biens sensibles	Confidentialité	Intégrité
Biens sensibles de l'utilisateur		
Eléments des clés d'accès manipulés par Zed !, en fonction des cas explicités plus haut : mot de passe ou code confidentiel éventuel, clé d'accès.	Forte	Forte
Fichiers et dossiers de l'utilisateur stockés dans les conteneurs	Forte	*N/A*
Biens sensibles de la TOE		
Fichiers de contrôle des conteneurs dont : les clés de chiffrement des fichiers	*Faible* Forte	Forte
Configuration de Zed! (policies)	*Faible*	Forte
Programmes de Zed !	*Faible*	Forte

Tableau 1 : Synthèse des biens sensibles

3.2. Hypothèses

Pour Zed!, 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 sans être observable directement et sans que cela puisse être intercepté (clavier sans fil…) par d'autres utilisateurs ou attaquants potentiels.

H.ENV_OPERATIONNEL L'environnement opérationnel ne permet pas à un attaquant d'accéder au disque (physiquement ou par le réseau) lorsque des données sensibles sont accessibles à un utilisateur légitime sur l'équipement. L'équipement doit donc apporter des protections efficaces contre l'accès illicite distant (pare-feu correctement configuré, antivirus et anti logiciels espions avec bases de données à jour etc.). Des mesures techniques ou organisationnelles doivent également empêcher un attaquant d'accéder au poste utilisateur en vue de modifier le logiciel de la TOE ou de récupérer directement les fichiers sensibles protégés par la TOE mais résidant en clair sur le poste.

H.CONFIANCE_ADM_TOE	Les administrateurs de la TOE sont des personnes de confiance. Ils sont formés à l'utilisation de la TOE tout comme les utilisateurs.
H.CONSERVATION_CLES	Les utilisateurs sont chargés de la conservation dans un lieu sûr et de la non divulgation des clés d'accès qui leurs ont été transmises par un administrateur. L'administrateur est également chargé de conservation dans un lieu sûr et de la non divulgation des clés de recouvrement.
H.CERTIFICATS	L'administrateur de la TOE est chargé de mettre en œuvre des procédures organisationnelles assurant la protection des certificats lors de leur remise aux utilisateurs. Il est également chargé, lors de la fourniture des clés d'accès possédant un certificat X509, de vérifier la validité de ces certificats et leur adéquation avec l'usage qui en est fait par la TOE.
H.ADMIN_WINDOWS	Les administrateurs Windows sont des personnes de confiance. Les administrateurs Windows de plus haut niveau du domaine Windows sont chargés d'interdire aux administrateurs Windows des sous-niveaux (domaines, contrôleurs de domaines, postes de travail) la modification des « policies » de la TOE. De même, les administrateurs et utilisateurs de la TOE ne doivent pas pouvoir modifier les « policies ». Les administrateurs de la sécurité de l'environnement Windows doivent garantir (ensemble si ils appartiennent à des organisations différentes) l'utilisation de politiques de sécurité conformes au niveau recherché (force des mots de passe notamment). Dans le cas contraire, les échanges doivent s'effectuer avec des conteneurs créés par chaque utilisateur et utilisés uniquement en émission vers les autres utilisateurs licites (un utilisateur ne doit alors pas accepter de recevoir un conteneur qu'il a lui-même créé).
H.CRYPTO_EXT	Les clés d'accès générées ou stockées à l'extérieur de la TOE doivent être conformes aux documents [CRYPTO_STD] et [CLES_STD].

3.3. Menaces [contre les biens sensibles de la TOE]

L'attaquant considéré est toujours externe. Il intercepte un courrier échangé contenant un conteneur en pièce jointe afin de l'attaquer en installant ou pas le produit Zed! (ou Zed! Edition Limitée) sur son poste. Par hypothèse, on considère que d'autres moyens sont utilisés pour protéger les données sensibles échangées et résidant sur le poste utilisateur (chiffrement du disque par exemple).

Il s'agit ici des menaces portant sur les biens sensibles de la TOE elle-même. Celles qui concernent les biens des utilisateurs sont couvertes par les Politiques de Sécurité Organisationnelles (services du produit) décrites plus loin.

L'attaquant considéré est doté d'un potentiel d'attaque « enhanced-basic » au sens des Critères Communs.

M.DETOURN_COMPOSANT	En possession d'un conteneur intercepté, un attaquant se procure le produit Zed! (ou Zed! Edition Limitée) et met en œuvre, éventuellement à bas niveau, les composants internes de la TOE, pour contourner certaines fonctions de sécurité. Il peut pour cela effectuer du «reverse-ingeniering» sur le programme, développer des programmes d'appel des fonctions internes de la TOE, agir sur la configuration interne de la TOE ou s'aider d'un debugger. Le bien impacté est le programme de la TOE (confidentialité et intégrité) ainsi que la configuration (intégrité). Il ne doit pas pouvoir, avec ces moyens, réussir à «pénétrer» le conteneur dans laquelle il n'aurait pas normalement accès.
M.ATTAQUE_ FIC_CONTROLE	Un attaquant récupère le fichier de contrôle de la TOE pour tenter de retrouver des informations protégées. Le bien impacté est donc le fichier de contrôle du conteneur (confidentialité). Par exemple, il modifie le fichier de contrôle de la TOE ou tente de retrouver des informations protégées à partir des fichiers chiffrées du conteneur et du fichier de contrôle de la TOE.
M.DETOURN_FIC_CONTROLE	Un attaquant récupère le fichier de contrôle de la TOE (stocké dans le conteneur) et le modifie afin de s'ajouter parmi les accès autorisés à ouvrir le conteneur (l'attaquant peut se positionner entre les deux correspondants par exemple). Le bien impacté est donc le fichier de contrôle du conteneur (intégrité). Il peut ainsi intercepter et lire les fichiers

échangés entre les correspondants légitimes ou envoyer le conteneur à un correspondant (en usurpant l'identité d'un utilisateur légitime) dans le but de lui faire envoyer des fichiers sensibles.

3.4. Politiques de sécurité de l'organisation

P.CONFIDENTIALITE	La TOE doit offrir un service de protection en confidentialité (chiffrement), automatique et systématique, du stockage et de la transmission (pièce jointe de courrier électronique) des fichiers sensibles des utilisateurs.
P.ACCES	La TOE doit permettre aux utilisateurs de fournir une clé d'accès permettant d'accéder aux fichiers sensibles du conteneur auquel ils désirent accéder. S'ils ne peuvent fournir une clé d'accès valide pour le conteneur, l'accès doit être rejeté.
P.RECOUVREMENT	La TOE doit offrir un service de recouvrement des fichiers sensibles des utilisateurs par l'emploi de clés d'accès de recouvrement gérées par l'administrateur de la TOE. Ces clés sont systématiquement et automatiquement affectées lors de l'initialisation des conteneurs. Cette politique ne concerne que Zed ! Edition standard.
P.ADMIN_ACCES	La TOE doit offrir un service de gestion des accès (Zed ! Edition standard seulement).
P.DCSSI	Le référentiel de l'ANSSI ([CRYPTO_STD] et [CLES_STD]) doit être suivi pour la gestion des clés et pour les mécanismes cryptographiques utilisés dans la TOE. Des mesures élémentaires doivent notamment permettre de supprimer les informations très sensibles telles que les clés susceptibles de résider en mémoire après utilisation. La mise en œuvre du référentiel de l'ANSSI doit apparaitre et se décliner dans les manuels d'utilisation de la TOE.

4. Objectifs de sécurité

4.1. Objectifs de sécurité pour la TOE

4.1.1. Contrôle d'accès

O.ACCES La TOE doit permettre de visualiser les accès et gérer les clés d'accès au conteneur (Zed ! Edition standard seulement).

O.AUTH La TOE doit permettre d'identifier et authentifier tout utilisateur.

Pour cela, la TOE ne doit autoriser l'accès à un fichier d'un conteneur qu'après présentation d'une clé d'accès valide pour ce conteneur.

O.ROLES La TOE doit gérer deux rôles d'utilisateurs pour un conteneur chiffrée : un rôle 'utilisateur normal' ou plus simplement 'utilisateur' (utilisation des fichiers du conteneur sous condition de présentation d'une clé d'accès valide) et un rôle 'administrateur' (installation, recouvrement, plus possibilité de gérer les clés d'accès).

4.1.2. Cryptographie

O.CHIFFREMENT La TOE doit chiffrer et déchiffrer les données sensibles par l'emploi de clés cryptographiques.

O.ALGO_STD La TOE doit produire des aléas et fournir un choix d'algorithmes cryptographiques et de tailles de clés conformes à l'état de l'art et aux standards de ce domaine, prévus dans [CRYPTO_STD] et complétés par [CLES_STD].

4.1.3. Gestion

O.RECOUVREMENT La TOE doit permettre d'affecter des clés d'accès de recouvrement (Zed ! Edition standard seulement).

4.1.4. Effacement

O.EFF_RESIDUS La TOE doit assurer le nettoyage des traces de données sensibles (clés) dans la mémoire (RAM) dès la fin des opérations réalisées par la TOE.

4.2. Objectifs de sécurité pour l'environnement

4.2.1. Utilisation

OE.NON_OBSERV L'environnement physique d'utilisation de la TOE doit permettre aux utilisateurs et aux administrateurs d'entrer leur mot de passe sans être observables directement ou sans que la saisie soit interceptable (clavier sans fil,…) par d'autres utilisateurs ou attaquants potentiels. Les mots de passe partagés entre les correspondants doivent être échangés à travers des canaux organisationnels protégés.

OE.ENV_OPERATIONNEL Lorsque l'utilisateur est authentifié, l'environnement opérationnel doit assurer la confidentialité des données sensibles et des données d'authentification.

Note d'application :
L'équipement doit apporter des protections efficaces contre l'écoute illicite et la transmission non autorisée de données (pare-feu correctement configuré, antivirus avec base de données à jour, « anti-spyware », etc.).
Les applications installées sur l'équipement ne doivent pas perturber le bon fonctionnement de la TOE.

	Des mesures techniques ou organisationnelles doivent également empêcher un attaquant d'accéder au poste utilisateur en vue de modifier le logiciel de la TOE ou de récupérer directement les fichiers sensibles protégés par la TOE mais résidant en clair sur le poste.
OE.SO_CONF	Les administrateurs de la TOE doivent être des personnes de confiance.
OE.CONSERV_CLES	Les utilisateurs doivent conserver, dans un lieu sûr, les clés d'accès qui leurs ont été transmises par un administrateur de la TOE et empêcher leur divulgation. L'administrateur de la TOE doit conserver ses clés de recouvrement dans un lieu sûr et empêcher leur divulgation.

4.2.2. Formation des utilisateurs

OE.FORMATION Les utilisateurs de la TOE doivent être formés à l'utilisation de la TOE et sensibilisés à la sécurité informatique (ceci prend en compte la sensibilisation sur la qualité des clés d'accès et de leur support lorsqu'elles sont hébergées par un porte-clés). Les administrateurs de la TOE doivent recevoir une formation adaptée à cette fonction.

OE.CRYPTO_EXT Les administrateurs de la TOE doivent être sensibilisés sur la qualité des clés d'accès qu'ils apportent à la TOE afin que ces clés soient conformes à l'état de l'art dans leur implémentation. Ils doivent également être sensibilisés à la qualité du support de ces clés lorsqu'elles sont hébergées par un porte-clés externe.

4.2.3. Administration

OE.CERTIFICATS L'administrateur de la TOE est chargé de mettre en œuvre des procédures organisationnelles assurant la protection des certificats lors de leur remise aux utilisateurs. Il est également chargé, lors de la fourniture des clés d'accès possédant un certificat X509, vérifier la validité de ces certificats et leur adéquation avec l'usage qui en est fait par la TOE. Cette exigence s'applique en particulier aux certificats racines dits «authenticode» à partir desquels la vérification d'intégrité de la TOE peut être effectuée.

OE.ADM_ROOT_WINDOWS Les administrateurs Windows sont des personnes de confiance.

Les administrateurs de plus haut niveau du domaine Windows doivent interdire aux administrateurs des sous-niveaux (domaines, contrôleurs de domaines, postes de travail) la modification des « policies » de la TOE. De même, les administrateurs de la TOE ne peuvent modifier les « policies ». En conséquence, ces administrateurs de plus haut niveau doivent eux-mêmes être des personnes de confiance.

Si les correspondants appartiennent à des entités

gérés par des administrateurs de la sécurité de l'environnement Windows différents, ceux-ci doivent garantir ensemble l'utilisation de politiques de sécurité conformes au niveau recherché (force des mots de passe notamment). Dans le cas contraire, les échanges doivent s'effectuer avec un conteneur créé par chaque utilisateur et utilisé uniquement en émission.

5. Exigences de sécurité des TI

5.1. Exigences de sécurité de la TOE

Dans cette section, les exigences de sécurité de la TOE ont été traduites en français afin d'améliorer leur compréhension. Le texte officiel servant de référence se trouve dans l'annexe A.

Toutes les opérations sur les composants (assignation, sélection, itération et raffinement) sont représentées par des caractères en italiques.

5.1.1. Exigences fonctionnelles de sécurité de la TOE

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

Composants CC retenus	
FCS_CKM.1	Génération des clés cryptographiques
FCS_CKM.3	Accès aux clés cryptographiques
FCS_CKM.4	Destruction des clés cryptographiques
FCS_COP.1	Opération cryptographique
FDP_ACC.1	Contrôle d'accès partiel
FDP_ACF.1	Contrôle d'accès basé sur les attributs de sécurité
FDP_ITC.1	Importation depuis une zone hors du contrôle de la TSF
FDP_RIP.1	Protection partielle des informations résiduelles
FIA_UAU.1	Programmation de l'authentification
FIA_UID.1	Programmation de l'identification
FMT_MOF.1	Administration des fonctions de la TSF
FMT_MSA.1	Gestion des attributs de sécurité
FMT_MSA.2	Attributs de sécurité sûrs
FMT_MSA.3	Initialisation statique d'attribut
FMT_MTD.1	Gestion des données de la TSF
FMT_SMF.1	Spécification des fonctions d'administration
FMT_SMR.1	Rôles de sécurité

Tableau 2 : Exigences fonctionnelles de sécurité pour la TOE

5.1.1.1 Introduction

Les exigences fonctionnelles de sécurité (SFR) font référence aux sujets suivants:

- Administrateurs et utilisateur de la TOE avec comme attributs de sécurité leur clé d'accès permettant ou non d'effectuer les opérations d'ouverture des conteneurs.

Les exigences fonctionnelles de sécurité (SFR) font référence aux objets suivants:

- Conteneurs manipulés par les utilisateurs de la TOE et qui contiennent les données sensibles des utilisateurs (fichiers, clés),
- clés cryptographiques.

Les exigences fonctionnelles de sécurité (SFR) font référence aux opérations suivantes:

- Ouverture en lecture et écriture des conteneurs
- Création et destruction d'un conteneur,
- renommage d'un conteneur
- destruction d'un fichier du conteneur
- désactivation, activation et modification du comportement des fonctions de chiffrement des conteneurs

5.1.1.2 Classe FCS : Support Cryptographique

FCS_CKM	**Gestion des clés cryptographiques**
FCS_CKM.1	Génération des clés cryptographiques
FCS_CKM.1.1	La TSF doit générer les clés cryptographiques conformément à un algorithme de génération de clés cryptographiques spécifié parmi les suivants *génération de nombres pseudo-aléatoires et diversification de clés* et à des tailles de clés cryptographiques *de clés symétriques de 128 à 256 bits* qui satisfont aux standards *PKCS#5 v2.0.* *Raffinement non éditorial :* *L'Edition limitée ne permet pas de générer de clé de chiffrement des fichiers (initialisation du conteneur).*
FCS_CKM.3	Accès aux clés cryptographiques
FCS_CKM.3.1	La TSF doit réaliser *l'utilisation de clés* conformément à une méthode d'accès aux clés cryptographiques spécifiée *par déchiffrement (déwrapping) des clés par la clé d'accès.*
FCS_CKM.4	Destruction de clés cryptographiques

FCS_CKM.4.1	La TSF doit détruire les clés cryptographiques conformément à une méthode de destruction spécifiée de clés cryptographiques *par réécriture de motifs composés de zéros*. *Raffinement non éditorial :* *L'Edition limitée ne permet pas de supprimer un accès et donc de détruire la clé associée.*

FCS_COP	**Opération cryptographique**
FCS_COP.1	Opération cryptographique
FCS_COP.1.1	La TSF doit exécuter le *hachage, le chiffrement, le déchiffrement, la génération de clés, le wrapping de clés et la dérivation de clés* conformément à un algorithme cryptographique spécifié *SHA-256, RSA, 3DES et AES* et avec des tailles de clés cryptographiques *de 128 à 256 bits pour les clés symétriques et de 1536 à 2048 bits pour les clés asymétriques* qui satisfont à ce qui suit: *FIPS 180-2 (SHA-256), ANSI X9.52-1998 (3DES), FIPS 197 (AES) et PKCS#1 (RSA).* *Raffinement non éditorial :* *L'Edition limitée ne permet pas de générer de clé.*

5.1.1.3 Classe FDP : Protection des données de l'utilisateur

FDP_ACC	**Politique de contrôle d'accès**
FDP_ACC.1	Contrôle d'accès partiel
FDP_ACC.1.1	La TSF doit appliquer la politique *SFP.ACCESS_OBJ* aux : *Sujets: Utilisateurs de la TOE* *Objets: Conteneurs contenant les fichiers utilisateur et le fichier de contrôle.*

FDP_ACF	**Fonctions de contrôle d'accès**
FDP_ACF.1	Contrôle d'accès basé sur les attributs de sécurité
FDP_ACF.1.1	La TSF doit appliquer la politique *SFP.ACCESS_OBJ* aux objets en fonction des : *Sujets : Utilisateurs de la TOE* *Attributs de sécurité: Clés d'accès utilisateur permettant ou non d'ouvrir le conteneur.*

FDP_ACF.1.2	La TSF doit appliquer les règles suivantes pour déterminer si une opération entre des sujets contrôlés et des objets contrôlés est autorisée : *Objet : Conteneur* *Opération: Ouverture en lecture et écriture du conteneur* *Règle : authentification réussie auprès du conteneur à l'aide de la clé d'accès utilisateur.*
FDP_ACF.1.3	La TSF doit autoriser explicitement l'accès de sujets à des objets en fonction des règles complémentaires suivantes : *Aucune*.
FDP_ACF.1.4	La TSF doit refuser explicitement l'accès de sujets à des objets en fonction de : *Aucune*.

FDP_ITC	**Importation depuis une zone hors du contrôle de la TSF**
FDP_ITC.1	Importation de données utilisateur sans attributs de sécurité
FDP_ITC.1.1	La TSF doit appliquer *les politiques de sécurité des fonctions (SFP) SFP.ACCESS_OBJ* lors de l'importation de données utilisateur, contrôlées par les SFP, en provenance de l'extérieur du TOE.
FDP_ITC.1.2	La TSF doit ignorer tout attribut de sécurité associé aux données utilisateur lorsqu'elles sont importées depuis l'extérieur du TOE.
FDP_ITC.1.3	La TSF doit appliquer les règles suivantes lors de l'importation des données utilisateur contrôlées par la SFP en provenance de l'extérieur du TOE : *Aucune*

FDP_RIP.1	**Protection des informations résiduelles**
FDP_RIP.1	Protection partielle des informations résiduelles
FDP_RIP.1.1	La TSF doit garantir que toute information contenue précédemment dans une ressource est rendue inaccessible lors de *la désallocation de la ressource* des objets suivants : *clés cryptographiques.*

5.1.1.4 Classe FIA : Identification et authentification

FIA_UAU	**Authentification de l'utilisateur**
FIA_UAU.1	Programmation de l'authentification
FIA_UAU.1.1	The TSF doit permettre *la création, le renommage et la destruction d'un container, la destruction d'un fichier du conteneur* pour le compte de l'utilisateur avant que celui-ci ne soit authentifié.
FIA_UAU.1.2	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.

FIA_UID	**Identification de l'utilisateur**
FIA_UID.1	Programmation de l'identification
FIA_UID.1.1	The TSF doit permettre *la création, le renommage et la destruction d'un container, la destruction d'un fichier du conteneur* pour le compte de l'utilisateur avant que celui-ci ne soit identifié.
FIA_UID.1.2	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.

5.1.1.5 Classe FMT : Administration de la sécurité

FMT_MOF	**Administration des fonctions de la TSF**
FMT_MOF.1	Administration du comportement des fonctions de sécurité
FMT_MOF.1.1	La TSF doit restreindre l'aptitude de *déterminer le comportement, désactiver, activer ou modifier le comportement* des fonctions *de chiffrement des conteneurs »* aux *administrateurs de la TOE*.

FMT_MSA	**Administration des attributs de sécurité**
FMT_MSA.1	Gestion des attributs de sécurité
FMT_MSA.1.1	La TSF doit mettre en œuvre la *politique SFP.ACCESS_ROLES* pour restreindre aux *utilisateurs et administrateurs de la TOE* la possibilité de *modifier ou supprimer* les *clés d'accès au conteneur*. *Raffinement non éditorial :* *Ce composant ne s'applique qu'à l'Edition Standard (l'Edition limitée ne permet pas de gérer les accès).*
FMT_MSA.2	Attributs de sécurité sûrs
FMT_MSA.2.1	La TSF doit garantir que seules des valeurs sûres sont acceptées pour les *clés d'accès au conteneur*. *Raffinement non éditorial :* *Ce composant ne s'applique qu'à l'Edition Standard (l'Edition limitée ne permet pas de créer des accès).*
FMT_MSA.3	Initialisation statique d'attribut
FMT_MSA.3.1	La TSF doit mettre en œuvre *la politique SFP.ACCESS_ROLES* afin de fournir des valeurs par défaut *restrictives* pour les attributs de sécurité qui sont utilisés pour appliquer la SFP.
FMT_MSA.3.2	La TSF doit permettre aux administrateurs de la TOE de spécifier des valeurs initiales alternatives pour remplacer les valeurs par défaut lorsqu'un objet ou une information est créé.

	Raffinement non éditorial : *Ce composant ne s'applique qu'à l'Edition Standard (l'Edition limitée ne permet pas de gérer les accès).*
FMT_MTD	**Gestion des données de la TSF**
FMT_MTD.1	Gestion des données de la TSF
FMT_MTD.1.1	La TSF doit restreindre, aux administrateurs de la TOE, la possibilité de *changer la valeur par défaut, interroger, modifier ou supprimer* les *stratégies de sécurité (policies)*.
FMT_SMF	**Spécification des fonctions d'administration**
FMT_SMF.1	Spécification des fonctions d'administration
FMT_SMF.1.1	La TSF doit être capable d'exécuter les fonctions d'administration suivantes : - *Les fonctions de gestion des accès* - *Les fonctions de gestion des conteneurs* - *La fonction de recouvrement* - *Les fonctions d'initialisation des paramètres utilisés par les fonctions de sécurité* *Raffinement non éditorial :* *L'Edition limité ne permet pas de gérer les accès (et donc d'affecter des clés de recouvrement).*
FMT_SMR	**Rôle pour l'administration de la sécurité**
FMT_SMR.1	Rôles de sécurité
FMT_SMR.1.1	La TSF doit tenir à jour les rôles *administrateur de la TOE et utilisateur de la TOE*.
FMT_SMR.1.2	La TSF doit être capable d'associer les utilisateurs aux rôles

5.1.2. Exigences d'assurance de sécurité de la TOE

Comme indiqué au paragraphe 3.3, la TOE doit être résistante aux attaques de pénétration effectuées par un attaquant ayant un potentiel d'attaque « enhanced-basic ».

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

EAL3 augmenté des composants ALC_FLR.3 et AVA_VAN.3 associé à une expertise de l'implémentation de la cryptographie décrite dans [QUALIF_STD].

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

Composant		Commentaire
ADV_ARC.1	Security architecture description	EAL3
ADV_FSP.3	Functional specification with complete summary	EAL3
ADV_TDS.2	Architectural design	EAL3
AGD_OPE.1	Operational user guidance	EAL3
AGD_PRE.1	Preparative procedures	EAL3
ALC_CMC.3	Authorisation controls	EAL3
ALC_CMS.3	Implementation representation CM coverage	EAL3
ALC_DEL.1	Delivery procedures	EAL3
ALC_DVS.1	Identification of security measures	EAL3
ALC_FLR.3	Systematic flaw remediation	+
ALC_LCD.1	Developer defined life-cycle model	EAL3
ASE_CCL.1	Conformance claims	EAL3
ASE_ECD.1	Extended components definition	EAL3
ASE_INT.1	ST introduction	EAL3
ASE_OBJ.2	Security objectives	EAL3
ASE_REQ.2	Security requirements	EAL3
ASE_SPD.1	Security problem definition	EAL3
ASE_TSS.1	TOE summary specification	EAL3
ATE_COV.2	Analysis of coverage	EAL3
ATE_DPT.1	Testing: basic design	EAL3
ATE_FUN.1	Functional testing	EAL3
ATE_IND.2	Independent testing - sample	EAL3
AVA_VAN.3	Focused vulnerability analysis	+

Tableau 3 : Composants d'assurance de sécurité

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

6. Spécifications globales de la TOE

6.1. Fonctions de sécurité de la TOE

Les fonctions de sécurité réalisées par la TOE sont décrites dans ce chapitre.

F.CONTROLE_ACCES **Contrôle d'accès au conteneur**

Cette fonction de sécurité constitue l'interface réalisant le contrôle d'accès obligatoire pour ouvrir le conteneur contrôlé par la TOE.

F.ENTREE_SECURISEE **Entrée sécurisée**

Cette fonction de sécurité recouvre la communication sécurisée de données fournies en entrée de la TOE (en utilisant pour cela des fonctions de chiffrement et déchiffrement de clé de conteneur).

F.GESTION_DROITS **Gestion des droits**

Cette fonction de sécurité gère les utilisateurs et les droits qui leur sont associés (on y distingue notamment l'accès de recouvrement). Un accès correspond à une clé d'accès (une clé cryptographique) que possède un utilisateur et permettant d'obtenir les éléments de chiffrement/déchiffrement de la zone.
Cette fonction de sécurité ne s'applique qu'à l'Edition Standard.

F.CONFIGURATION_TOE **Modification de la configuration de la TOE**

Cette fonction de sécurité couvre l'ensemble des opérations de configuration de la TOE (et assure que seules des valeurs sûres de paramètres de configuration peuvent être utilisées). La TOE ne peut pas fonctionner sans être configurée. Les données de configuration concernent ici les «« policies »» de Windows exploitées par la TOE.

F.GESTION_CLES **Gestion des clés**

Cette fonction de sécurité réalise les opérations de création, de suppression des clés de conteneur ainsi que les opérations d'accès à ces clés.

L'Edition limitée n'est concernée que par l'accès aux clés.

F.GESTION_CONTENEUR **Opération sur les conteneurs**

Cette fonction de sécurité constitue le point d'entrée des opérations sur les conteneurs (création, initialisation, suppression, affichage, nommage).

F.OPERATIONS_CRYPTO Implémentation des opérations cryptographiques

Cette fonction de sécurité couvre l'ensemble des opérations cryptographiques mises au service des autres fonctions de sécurité.

Certaines opérations cryptographiques (génération des clés) ne concernent pas l'Edition limitée.

6.2. Mesures d'assurance

Cette section décrit les mesures d'assurance qui sont mises en œuvre afin de satisfaire aux exigences d'assurance visées par l'évaluation de la TOE.

6.2.1. Mesures de l'environnement de développement

6.2.1.1 MA.ENV_CONF : Méthodes et outils de gestion de configuration

Le système de gestion de configuration couvre la gestion et le contrôle du développement, de la production et de la maintenance du logiciel Zed !. Son application permet d'affecter un identifiant unique à chaque version de la TOE et d'établir une liste des versions des composants qui constituent une version donnée.

Les procédures du système de gestion de configuration sont documentées et fournissent une liste de configuration pour la TOE.

Ces procédures documentent également le modèle de cycle de vie régissant le développement et la maintenance de la TOE.

Références des fournitures : ***A compléter***.

Argumentaire : ces procédures satisfont de manière directe aux exigences ALC_CMC.3, ALC_CMS.3 et ALC_LCD.1.

6.2.1.2 MA.ENV_SEC : Sécurité de l'environnement de développement

Les mesures de sécurité appliquées pour le développement et la maintenance du logiciel Zed ! garantissent l'intégrité du code exécutable de la TOE et la confidentialité des documents de développement associés.

Les mesures de sécurité de l'environnement de développement sont documentées, elles identifient précisément le périmètre de cet environnement et fournissent des traces de l'application de ces mesures.

Références des fournitures : ***A compléter***.

Argumentaire : ces procédures satisfont de manière directe à l'exigence ALC_DVS.1.

6.2.1.3 MA.ENV_LIV : Procédures de livraison

Les procédures et mesures mises en place pour transférer le logiciel Zed ! du développeur chez l'utilisateur final garantissent l'authenticité et l'intégrité de la TOE lors du transfert.

Les procédures de livraison sont documentées.

Références des fournitures : ***A compléter***.

Argumentaire : ces procédures satisfont de manière directe à l'exigence ALC_DEL.1.

6.2.1.4 MA.ENV_SUP : Procédures de correction des anomalies

Des procédures de correction des anomalies sont mises en place au niveau du laboratoire et du service support pour assurer une gestion et un contrôle des anomalies de sécurité découvertes en interne ou soumises par les exploitants, ainsi que la distribution des correctifs associés, une fois les anomalies résolues.

Les procédures visant à la correction des anomalies sont documentées, et les documents donnant des lignes directrices aux exploitants pour soumettre les anomalies sont fournit.

Références des fournitures : ***A compléter***.

Argumentaire : ces procédures satisfont de manière directe à l'exigence ALC_FLR.3.

6.2.2. MA.DEV : Documentation et outils de développement des fonctions de sécurité

Les documents permettant d'assurer un niveau de qualité compatible avec les exigences liées au paquet d'assurance sécurité sont fournit : spécifications fonctionnelles, architecture de sécurité, conception de haut niveau et, uniquement pour les fonctions cryptographiques, conception de bas niveau, documentation des outils et techniques de développement (compilateurs, makefiles, …) et code source. Ces documents forment les niveaux successifs de représentation de la fonctionnalité de sécurité.

Des correspondances entre ces niveaux sont établies, en commençant par les fonctions de sécurité des TI spécifiées de manière abrégée dans ce document.

Références des fournitures : ***A compléter***.

Argumentaire : ces mesures satisfont de manière directe aux exigences ADV_FSP.3, ADV_ARC.1 et ADV_TDS.2.

6.2.3. Test des fonctions de sécurité

6.2.3.1 MA.TEST_DEV : Procédure de test du développeur

Les documents produits à l'occasion des tests effectués sur la TOE sont fournit. Ces documents doivent décrire le plan et les procédures de tests suivies et montrer le degré de couverture des spécifications fonctionnelles par les tests. Ils doivent inclure les résultats effectifs des tests et démontrer que les fonctions de sécurité se comportent bien de la manière spécifiée dans les spécifications fonctionnelles.

Une TOE se prêtant au repassage des tests effectués est mise à disposition de l'évaluateur.

Références des fournitures : ***A compléter***.

Argumentaire : ces procédures satisfont de manière directe aux exigences ATE_FUN.1, ATE_COV.2, ATE_DPT.1 et à une partie d'ATE_IND.2 (repassage des tests).

6.2.3.2 MA.TEST_EVAL : Test indépendant

Le commanditaire met à disposition de l'évaluateur une TOE se prêtant à l'exécution de tests indépendants.

Références des fournitures : sans objet.

Argumentaire : ces mesures satisfont de manière directe à l'exigence ATE_IND.2 (test indépendant).

6.2.4. Documentation d'exploitation

6.2.4.1 MA.GUIDE_INST : Procédures d'installation et de démarrage

Ces procédures permettent l'installation et le démarrage de la TOE dans des conditions qui garantissent une exécution satisfaisante de ses fonctions de sécurité.

Afin de prévenir les risques d'utilisation impropre, la documentation d'installation et de démarrage doit spécifiquement identifier tous les modes d'exécution possibles de la TOE ainsi que leur impact sur la sécurité. Elle doit être claire, complète, cohérente, et accessible à l'audience visée. Elle doit enfin énumérer toutes les hypothèses relatives à l'environnement d'exploitation prévu et les exigences sur les mesures de sécurité (TI ou non-TI) qui doivent être présentes dans l'environnement.

Les procédures d'installation et de démarrage sûrs de la TOE sont documentées.

Références des fournitures : ***A compléter***.

Argumentaire : ces mesures satisfont de manière directe à l'exigence AGD_PRE.1.

6.2.4.2 MA.GUIDE_ADMIN : Documentation d'administration

La documentation d'exploitation à destination des administrateurs doit décrire le comportement des fonctions de sécurité et refléter les hypothèses sur l'environnement d'exploitation, dans une optique de configuration, de maintenance et de maintien en condition opérationnelle corrects des fonctions de sécurité. Elle doit également décrire les différents types d'événements relatifs à la sécurité susceptibles de survenir, et fournir des lignes directrices sur la manière de les prendre en compte.

Des exigences spécifiques à la prévention de l'utilisation impropre, similaires à celles sur la documentation d'installation et de démarrage, pèsent également sur la documentation d'administration La documentation d'administration est fournit.

Références des fournitures : ***A compléter***.

Argumentaire : ces mesures satisfont de manière directe à l'exigence AGD_OPE.1.

6.2.4.3 MA.GUIDE_UTILIS : Documentation utilisateur

La documentation d'exploitation à destination des utilisateurs doit décrire le comportement des fonctions de sécurité qu'ils ont besoin de connaître, et refléter les hypothèses sur l'environnement d'exploitation et les responsabilités qui les concernent (et notamment les situations qui nécessitent d'en référer à l'administrateur).

Des exigences spécifiques à la prévention de l'utilisation impropre, similaires à celles sur la documentation d'installation et de démarrage et d'administration, peuvent

également peser sur la documentation utilisateur, sous réserve de leur pertinence et de leur applicabilité aux utilisateurs.

La documentation utilisateur est fournit.

Références des fournitures : ***A compléter***.

Argumentaire : mesures satisfont de manière directe à l'exigence AGD_OPE.1.

6.2.5. MA.VUL : Estimation de la vulnérabilité

Le commanditaire met à disposition de l'évaluateur une TOE se prêtant à l'analyse de vulnérabilité effectuée par l'évaluateur.

Références des fournitures : sans objet.

Argumentaire : ces mesures satisfont de manière directe à l'exigence AVA_VAN.3.

6.2.6. MA.CRYPTO : Expertise cryptographique

Tout algorithme, mode opératoire ou protocole cryptographique rendant un service de sécurité décrit dans la cible de sécurité doit faire l'objet d'une analyse réalisée par l'ANSSI, en vue d'attester de son respect du référentiel technique cryptographique, conformément à [QUALIF_STD]. Il en est de même pour la génération d'aléa et la gestion des clés utilisées dans des mécanismes cryptographiques.

Le développeur doit donc fournir les éléments requis pour l'analyse des mécanismes cryptographiques, conformément à [QUALIF_STD].

Références des fournitures : ***A compléter***.

Argumentaire : ces mesures satisfont de manière directe à la conformité au processus de qualification de niveau standard défini par l'ANSSI dans [QUALIF_STD].

7. Annonces de conformité à un PP

Cette cible de sécurité ne déclare aucune conformité à un Profil de Protection.

8. Argumentaire

8.1. Argumentaire pour les objectifs de sécurité

Cette section présente les liens de couverture entre les objectifs de sécurité et les éléments qui constituent la définition de l'environnement de la TOE (hypothèses, politiques de l'organisation et menaces).

8.1.1. Hypothèses

Le tableau ci-dessous présente la couverture des hypothèses retenues par les objectifs de sécurité :

		OE.NON_OBSERV	OE.ENV_OPERATIONNEL	OE.SO_CONF	OE.CONSERV_CLES	OE.CERTIFICATS	OE.ADM_ROOT_WINDOWS	OE.FORMATION	OE.CRYPTO_EXT
	H.NON_OBSERV	X							
	H.ENV_OPERATIONNEL		X						
	H.CONFIANCE_ADM_TOE			X				X	
	H.CONSERVATION_CLES				X			X	
	H.CERTIFICATS					X			
	H.ADMIN_WINDOWS						X		
	H.CRYPTO_EXT								X

H.NON_OBSERV	L'environnement physique de la TOE permet aux utilisateurs d'entrer leur mot de passe sans être observable directement et sans que cela puisse être intercepté (clavier sans fil…) par d'autres utilisateurs ou attaquants potentiels.

L'objectif OE.NON_OBSERV couvre directement cette hypothèse en mettant à disposition de l'utilisateur un environnement adéquat.

H.ENV_OPERATIONNEL	L'environnement opérationnel ne permet pas à un attaquant d'accéder au disque lorsque des données sensibles sont accessibles à un utilisateur légitime sur l'équipement.

L'objectif OE.ENV_OPERATIONNEL couvre directement cette hypothèse en mettant à disposition de l'utilisateur un environnement opérationnel adéquat.

H.CONFIANCE_ADM_TOE	Les administrateurs de la TOE sont des personnes de confiance. Ils sont formés à l'utilisation de la TOE tout comme les utilisateurs.

Les objectifs OE.SO_CONF et OE.FORMATION couvrent directement cette hypothèse en employant des personnes de confiance et en leur apportant la formation nécessaire.

H.CONSERVATION_CLES	Les utilisateurs sont chargés de la conservation dans un lieu sûr et de la non divulgation des clés d'accès qui leurs ont été transmises par un administrateur. L'administrateur est également chargé de conservation dans un lieu sûr et de la non divulgation des clés de recouvrement.

Les objectifs OE.CONSERV_CLES et OE.FORMATION couvrent cette hypothèse en responsabilisant et en sensibilisant les utilisateurs et les administrateurs.

H.CERTIFICATS	L'administrateur de la TOE est chargé de mettre en œuvre des procédures organisationnelles assurant la protection des certificats lors de leur remise aux utilisateurs. Il est également chargé, lors de la fourniture des clés d'accès possédant un certificat X509, de vérifier la validité de ces certificats et leur adéquation avec l'usage qui en est fait par la TOE.

L'objectif OE.CERTIFICATS couvre directement cette hypothèse.

H.ADMIN_WINDOWS	Les administrateurs Windows sont des personnes de confiance. Les administrateurs Windows de plus haut niveau du domaine Windows sont chargés d'interdire aux administrateurs Windows des sous-niveaux (domaines, contrôleurs de domaines, postes de travail) la modification des « policies » de la TOE. De même, les administrateurs et utilisateurs de la TOE ne doivent pas pouvoir modifier les « policies ». Les administrateurs de la sécurité de l'environnement Windows doivent garantir (ensemble si ils appartiennent à des organisations différentes) l'utilisation de politiques de sécurité conformes au niveau recherché (force des mots de passe notamment). Dans le cas contraire, les échanges doivent s'effectuer avec des conteneurs créés par chaque utilisateur et utilisés uniquement en émission vers les autres utilisateurs licites (un utilisateur ne doit alors pas accepter de recevoir un conteneur qu'il a lui-même créé).

L'objectif OE.ADM_ROOT_WINDOWS couvre directement cette hypothèse.

H.CRYPTO_EXT	Les clés d'accès générées ou stockées à l'extérieur de la TOE doivent être conformes aux documents [CRYPTO_STD] et [CLES_STD].

L'objectif OE.CRYPTO_EXT couvre directement cette hypothèse.

8.1.2. Menaces

Le tableau ci-dessous présente les liens de couverture entre les objectifs de sécurité et les menaces retenues :

		O.CHIFFREMENT	O.ACCES	O.ALGO_STD	O.ROLES	O.EFF_RESIDUS	O.RECOUVREMENT	O.AUTH
Menaces	M.DETOURN_COMPOSANT			X		X		X
	M.ATTAQUE_FIC_CONTROLE			X				X
	M.DETOURN_FIC_CONTROLE			X				X

Tableau 4 : Couverture des menaces par les objectifs de sécurité

M.DETOURN_COMPOSANT

En possession d'un conteneur intercepté, un attaquant se procure le produit Zed! (ou Zed! Edition Limitée) et met en œuvre, éventuellement à bas niveau, les composants internes de la TOE, pour contourner certaines fonctions de sécurité. Il peut pour cela effectuer du «reverse-ingeniering» sur le programme, développer des programmes d'appel des fonctions internes de la TOE, ou s'aider d'un debugger. Le bien impacté est le programme de la TOE (confidentialité et intégrité).

Il ne doit pas pouvoir, avec ces moyens, réussir à «pénétrer» le conteneur dans lequel il n'aurait pas normalement accès.

➔ Pour prévenir cette menace, la TOE doit :

- Garantir le fait qu'il n'est pas possible, cryptographiquement, de retrouver les clés de chiffrement du conteneur sans fournir une clé d'accès valide : le détournement d'un composant (i.e. sa mise en œuvre de façon détournée ou non prévue) ne peut pas permettre de franchir cette barrière (O.AUTH et O.ALGO_STD),

- Garantir le fait qu'avant toute opération sur la TOE, une authentification est nécessaire (O.AUTH),
- Garantir le fait qu'un composant détourné ne conserve pas de résidus de clé permettant de présenter un chemin pour une attaque (O.EFF_RESIDUS).

➔ Pour détecter l'occurrence de la menace, la TOE doit :

rien

➔ Pour limiter l'impact de la menace, la TOE doit :

Rien

M.ATTAQUE_FIC_CONTROLE	Un attaquant récupère le fichier de contrôle de la TOE pour tenter de retrouver des informations protégées. Le bien impacté est donc le fichier de contrôle du conteneur (confidentialité). Par exemple, il modifie le fichier de contrôle de la TOE ou tente de retrouver des informations protégées à partir des fichiers chiffrées du conteneur et du fichier de contrôle de la TOE.

➔ Pour prévenir cette menace, la TOE doit :

- Garantir le fait qu'il n'est pas possible, cryptographiquement, de retrouver les clés de chiffrement du conteneur sans fournir une clé d'accès valide, et par le fait que cet objectif prévoit que le fichier de contrôle de la TOE respecte également ce principe (O.AUTH et O.ALGO_STD).
- Garantir le fait que les fichiers internes des différents conteneurs sont rendus «cryptographiquement différents» par l'utilisation d'aléas ne permettant pas de tirer des enseignements d'un fichier de contrôle pour en attaquer un autre (O.ALGO_STD).

➔ Pour détecter l'occurrence de la menace, la TOE doit :

rien

➔ Pour limiter l'impact de la menace, la TOE doit :

Rien

M.DETOURN_FIC_CONTROLE	Un attaquant récupère le fichier de contrôle de la TOE (stocké dans le conteneur) et le modifie afin de s'ajouter parmi les accès autorisés à ouvrir le conteneur (l'attaquant peut se positionner entre les deux correspondants par exemple). Le bien impacté est donc le fichier de contrôle du

conteneur (intégrité).

Il peut ainsi intercepter et lire les fichiers échangés entre les correspondants légitimes ou envoyer le conteneur à un correspondant (en usurpant l'identité d'un utilisateur légitime) dans le but de lui faire envoyer des fichiers sensibles.

➔ Pour prévenir cette menace, la TOE doit :

- Garantir le fait qu'il n'est pas possible, cryptographiquement, de retrouver les clés de chiffrement du conteneur sans fournir une clé d'accès valide : le détournement du fichier de contrôle (i.e. sa mise en œuvre de façon détournée ou non prévue) est interdit par cette barrière (O.AUTH et O.ALGO_STD),
- Garantir le fait qu'avant toute opération sur la TOE, une authentification est nécessaire (O.AUTH),

➔ Pour détecter l'occurrence de la menace, la TOE doit :

rien

➔ Pour limiter l'impact de la menace, la TOE doit :

Rien

8.1.3. Politiques de sécurité de l'organisation

Le tableau ci-dessous présente les liens de couverture entre les objectifs de sécurité et les politiques de sécurité de l'organisation retenues :

		O.CHIFFREMENT	O.ACCES	O.ALGO_STD	O.ROLES	O.EFF_RESIDUS	O.RECOUVREMENT	O.AUTH
	P.CONFIDENTIALITE	X		X		X		
	P.ACCES					X		X
	P.RECOUVREMENT				X	X	X	X
	P.ADMIN_ACCES		X			X		X
	P.DCSSI			X		X		

Tableau 5 : Couverture des politiques de sécurité de l'organisation par les objectifs de sécurité

P.CONFIDENTIALITE La TOE doit offrir un service de protection en confidentialité (chiffrement), automatique et systématique, du stockage et de la transmission (pièce jointe de courrier électronique) des fichiers sensibles des utilisateurs.

Note: cette politique concerne la création initiale du conteneur, et le fait qu'une fois le conteneur créée, tout fichier déposé dans celui ci est stocké chiffré. Cette politique ne concerne pas les accès au conteneur, qui relèvent de P.ACCES.

Pour mettre en œuvre la politique, la TOE :

- Génère, lors du chiffrement, des aléas pour la création des clés de chiffrement du conteneur (O.ALGO_STD) ;
- Chiffre les fichiers dans le conteneur (O.CHIFFREMENT) ;
- Efface les traces mémoire liées aux clés de chiffrement des fichiers (O.EFF_RESIDUS) ;

P.ACCES La TOE doit permettre aux utilisateurs de fournir une clé d'accès permettant d'accéder aux fichiers sensibles du conteneur auquel ils désirent accéder. S'ils ne peuvent fournir une clé d'accès valide pour le conteneur, l'accès doit être rejeté.

Note: cette politique ne concerne pas la gestion des accès (création et suppression assurées par P.ADMIN_ACCES), mais l'utilisation d'un accès.

Pour mettre en œuvre la politique, la TOE :

- Demande une authentification avant toute manipulation du fichier d'un conteneur (O.AUTH);
- Fait en sorte que seule une clé d'accès valide puisse permettre de retrouver les clés de chiffrement du conteneur, et que les fichiers ou informations internes de la TOE ne permettent pas de faire autrement (O.AUTH).
- Efface les traces mémoire liées aux éventuels calculs cryptographiques intermédiaires (dérivation de mots de passe) ou au transport des valeurs de clés de chiffrement lorsqu'elles sont calculées par un dispositif cryptographique externe (token) (O.EFF_RESIDUS) ;

P.RECOUVREMENT	La TOE doit offrir un service de recouvrement des fichiers sensibles des utilisateurs par l'emploi de clés d'accès de recouvrement gérées par l'administrateur de la TOE. Ces clés sont systématiquement et automatiquement affectées lors de l'initialisation des conteneurs. Cette politique ne concerne que Zed ! Edition standard.

Pour mettre en œuvre la politique, la TOE :

- Demande une authentification pour accéder à la gestion des clés de recouvrement (O.AUTH) ;
- N'autorise que l'administrateur à effectuer les opérations de recouvrement (O.ROLES)
- Fait en sorte que seule une clé d'accès valide puisse permettre de retrouver les clés de chiffrement d'un conteneur, et que les fichiers ou informations internes de la TOE ne permettent pas de faire autrement (O.AUTH) ;
- Permet d'affecter des clés d'accès de recouvrement au conteneur (O.RECOUVREMENT).
- Efface les traces mémoire liées aux éventuels calculs cryptographiques intermédiaires (dérivation de mots de passe) ou au transport des valeurs de clés de chiffrement lorsqu'elles sont calculées par un dispositif cryptographique externe (token) (O.EFF_RESIDUS).

P.ADMIN_ACCES	La TOE doit offrir un service de gestion des accès (Zed ! Edition standard seulement).

Pour mettre en œuvre la politique, la TOE :

- Demande une authentification avant de permettre la gestion des accès au conteneur chiffré (O.AUTH) ;
- Permet de visualiser les accès et gérer les clés d'accès au conteneur (O.ACCES).
- Efface les traces mémoires des clés de chiffrement manipulées (O.EFF_RESIDUS).

P.DCSSI	Le référentiel de l'ANSSI ([CRYPTO_STD] et [CLES_STD]) doit être suivi pour la gestion des clés et pour les mécanismes cryptographiques utilisés dans la TOE.

➔ Pour mettre en œuvre la politique, la TOE :

- Fournit un choix d'algorithmes cryptographiques et de tailles de clés conformes à l'état de l'art et aux standards de ce domaine, prévus dans [CRYPTO_STD] et complétés par [CLES_STD] pour la gestion des clés.
- Génère des aléas de qualité cryptographique (O.ALGO_STD),
- Efface les traces mémoires des clés de chiffrement manipulées (O.EFF_RESIDUS).

8.2. Argumentaire pour les exigences de sécurité

8.2.1. Dépendances entre exigences fonctionnelles de sécurité

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

Composant	Dépendances	Dépendances satisfaites
FCS_CKM.1	[FCS_CKM.2 ou FCS_COP.1], FCS_CKM.4	FCS_COP.1, FCS_CKM.4
FCS_CKM.3	[FDP_ITC.1 ou FDP_ITC.2 ou FCS_CKM.1], FCS_CKM.4	FDP_ITC.1, FCS_CKM.1, FCS_CKM.4
FCS_CKM.4	[FDP_ITC.1 ou FDP_ITC.2 ou FCS_CKM.1]	FDP_ITC.1, FCS_CKM.1
FCS_COP.1	[FDP_ITC.1 ou FDP_ITC.2 ou FCS_CKM.1], FCS_CKM.4	FDP_ITC.1, FCS_CKM.1, FCS_CKM.4
FDP_ACC.1	FDP_ACF.1	FDP_ACF.1
FDP_ACF.1	FDP_ACC.1, FMT_MSA.3	FDP_ACC.1, FMT_MSA.3
FDP_ITC.1	[FDP_ACC.1 ou FDP_IFC.1], FMT_MSA.3	FDP_ACC.1, FMT_MSA.3
FDP_RIP.1	Aucune	Aucune
FIA_UAU.1	FIA_UID.1	FIA_UID.1
FIA_UID.1	Aucune	Aucune
FMT_MOF.1	FMT_SMF.1, FMT_SMR.1	FMT_SMF.1, FMT_SMR.1
FMT_MSA.1	[FDP_ACC.1 ou FDP_IFC.1], FMT_SMF.1, FMT_SMR.1	FDP_ACC.1, FMT_SMF.1, FMT_SMR.1
FMT_MSA.2	[FDP_ACC.1 ou FDP_IFC.1], FMT_MSA.1, FMT_SMR.1	FDP_ACC.1, FMT_MSA.1, FMT_SMR.1
FMT_MSA.3	FMT_MSA.1, FMT_SMR.1	FMT_MSA.1, FMT_SMR.1
FMT_MTD.1	FMT_SMR.1, FMT_SMF.1	FMT_SMR.1, FMT_SMF.1
FMT_SMF.1	Aucune	Aucune
FMT_SMR.1	FIA_UID.1	FIA_UID.1

Tableau 6 : Satisfaction des dépendances entre exigences fonctionnelles de sécurité

8.2.2. Dépendances entre exigences d'assurance de sécurité

Le tableau ci-dessous présente la couverture des dépendances entre les composants d'assurance sélectionnés :

Composant	Dépendances	Dépendances satisfaites
ADV_ARC.1	ADV_FSP.1, ADV_TDS.1	ADV_FSP.3, ADV_TDS.2
ADV_FSP.3	ADV_TDS.1	ADV_TDS.2
ADV_TDS.2	ADV_FSP.3	ADV_FSP.3
AGD_OPE.1	ADV_FSP.1	ADV_FSP.3
AGD_PRE.1	Aucune	Aucune
ALC_CMC.3	ALC_CMS.1, ALC_DVS.1, ALC_LCD.1	ALC_CMS.3, ALC_DVS.1, ALC_LCD.1
ALC_CMS.3	Aucune	Aucune
ALC_DEL.1	Aucune	Aucune
ALC_DVS.1	Aucune	Aucune
ALC_FLR.3	Aucune	Aucune
ALC_LCD.1	Aucune	Aucune
ASE_CCL.1	ASE_INT.1, ASE_ECD.1, ASE_REQ.1	ASE_INT.1, ASE_ECD.1, ASE_REQ.2
ASE_ECD.1	Aucune	Aucune
ASE_INT.1	Aucune	Aucune
ASE_OBJ.2	ASE_SPD.1	ASE_SPD.1
ASE_REQ.2	ASE_OBJ.2, ASE_ECD.1	ASE_OBJ.2, ASE_ECD.1
ASE_SPD.1	Aucune	Aucune
ASE_TSS.1	ASE_INT.1, ASE_REQ.1, ADV_FSP.1	ASE_INT.1, ASE_REQ.2, ADV_FSP.3
ATE_COV.2	ADV_FSP.2, ATE_FUN.1	ADV_FSP.3, ATE_FUN.1
ATE_DPT.1	ADV_ARC.1, ADV_TDS.2, ATE_FUN.1	ADV_ARC.1, ADV_TDS.2, ATE_FUN.1
ATE_FUN.1	ATE_COV.1	ATE_COV.2
ATE_IND.2	ADV_FSP.2, AGD_OPE.1, AGD_PRE.1, ATE_COV.1, ATE_FUN.1	ADV_FSP.3, AGD_OPE.1, AGD_PRE.1, ATE_COV.2, ATE_FUN.1
AVA_VAN.3	ADV_ARC.1, ADV_FSP.2, ADV_TDS.3*, ADV_IMP.1*, AGD_OPE.1, AGD_PRE.1	ADV_ARC.1, ADV_FSP.3, AGD_OPE.1, AGD_PRE.1

Tableau 7 : Satisfaction des dépendances entre exigences d'assurance de sécurité

8.2.3. Argumentaire pour les dépendances non satisfaites

La dépendance AVA_VAN.3 avec ADV_IMP.1 et ADV_TDS.3 est implicitement réalisée au titre de l'expertise cryptographique menée dans le cadre du projet d'évaluation (le focus étant mis, à la demande de l'ANSSI, sur la conception de la partie traitant des mécanismes cryptographiques).

8.2.4. Argumentaire de couverture des objectifs de sécurité par les exigences fonctionnelles

Les tableaux ci-dessous présentent la couverture des composants fonctionnels sélectionnés par les objectifs de sécurité :

Objectifs de sécurité de la TOE	FCS_CKM.1	FCS_CKM.3	FCS_CKM.4	FCS_COP.1	FDP_ACC.1	FDP_ACF.1	FDP_ITC.1	FDP_RIP.1	FIA_UAU.1	FIA_UID.1	FMT_MOF.1	FMT_MSA.1	FMT_MSA.2	FMT_MSA.3	FMT_MTD.1	FMT_SMF.1	FMT_SMR.1
O.ACCES												X	X	X			
O.AUTH					X	X	X		X	X							
O.ROLES					X	X					X				X	X	X
O.CHIFFREMENT	X	X		X													
O.ALGO_STD	X	X	X	X													
O.RECOUVREMENT															X	X	X
O.EFF_RESIDUS								X									

Tableau 8 : Couverture des objectifs de sécurité par les exigences fonctionnelles de sécurité

8.2.4.1 Contrôle d'accès

O.ACCES La TOE doit permettre de visualiser les accès et gérer les clés d'accès au conteneur (Zed ! Edition standard seulement).

Afin de remplir cet objectif :

- La TOE assure que seuls les administrateurs et les utilisateurs licites de la TOE peuvent gérer les accès aux conteneurs (FMT_MSA.1).
- La TOE garantie, de plus, que seuls des valeurs sûres sont acceptées pour les attributs de sécurité (FMT_MSA.2).
- L'administrateur peut aussi définir les données statiques d'un conteneur, tel que la force des mots de passe, le type et la longueur des clés utilisées, les cartes et token supportés, certaines caractéristiques des certificats (FMT_MSA.3).

O.AUTH	La TOE doit permettre d'identifier et authentifier tout utilisateur.

Afin de remplir cet objectif :

- La TOE identifie et authentifie chaque utilisateur avant d'ouvrir les conteneurs chiffrés (FIA_UID.1 et FIA_UAU.1).
- Pour que la TOE donne l'accès à une zone chiffré, l'utilisateur doit présenter sa clé d'accès (token USB par exemple) en vue de son authentification (FDP_ITC.1). La TOE applique ensuite une politique de contrôle d'accès au conteneur (FDP_ACC.1) et basé sur les attributs de sécurité (FDP_ACF.1).

O.ROLES	La TOE doit gérer deux rôles d'utilisateurs pour un conteneur chiffrée : un rôle 'utilisateur normal' ou plus simplement 'utilisateur' (utilisation des fichiers du conteneur sous condition de présentation d'une clé d'accès valide) et un rôle 'administrateur' (installation, recouvrement, plus possibilité de gérer les clés d'accès).

Afin de remplir cet objectif :

- La TOE doit gérer et distinguer les rôles d'administrateur de la TOE et d'utilisateur de la TOE (FMT_SMR.1) et restreindre certaines fonctions aux administrateurs (FMT_MOF.1).
- La TOE permet aussi de contrôler l'accès des utilisateurs aux conteneurs et aux opérations sur ces conteneurs (FDP_ACC.1), et de restreindre l'accès aux seuls utilisateurs possédant la clé d'accès associée (FDP_ACF.1).
- Enfin, la TOE doit permettre de restreindre aux administrateurs certaines fonctions d'administration de la sécurité (FMT_SMF.1) et la gestion des « policies » (FMT_MTD.1).

8.2.4.2 Cryptographie

O.CHIFFREMENT	La TOE doit chiffrer et déchiffrer les données sensibles par l'emploi de clés cryptographiques.

Afin de remplir cet objectif :

- Pour chiffrer les fichiers présents dans un conteneur, la TOE doit tout d'abord être capable de générer les clés cryptographiques (FCS_CKM.1) et y accéder de manière sécurisée (FCS_CKM.3), afin de les utiliser pour réaliser les opérations cryptographiques selon différents algorithmes (FCS_COP.1).

O.ALGO_STD	La TOE doit produire des aléas et fournir un choix d'algorithmes cryptographiques et de tailles de clés conformes à l'état de l'art et aux standards de ce domaine, prévus dans [CRYPTO_STD] et complétés par [CLES_STD].

Afin de remplir cet objectif :

- La TOE doit être capable de fournir un choix d'algorithmes de génération (FCS_CKM.1), d'accès (FCS_CKM.3) et de destruction (FCS_CKM.4) de clés cryptographiques.
- Elle doit aussi permettre d'exécuter des opérations cryptographiques conformément à des algorithmes et tailles de clés cryptographique spécifiés (FCS_COP.1).

8.2.4.3 Gestion

O.RECOUVREMENT	La TOE doit permettre d'affecter des clés d'accès de recouvrement (Zed ! Edition standard seulement).

Afin de remplir cet objectif :

- La TOE doit permettre de restreindre aux administrateurs (rôle défini dans FMT_SMR.1) les fonctions de recouvrement (FMT_SMF.1) configurées dans les « policies » (FMT_MTD.1).

8.2.4.4 Effacement

O.EFF_RESIDUS	La TOE doit assurer le nettoyage des traces de données sensibles (clés) dans la mémoire (RAM) dès la fin des opérations réalisées par la TOE.

Afin de remplir cet objectif :

- La TOE permet un nettoyage totalement sécurisé des clés dans la mémoire (RAM) (FDP_RIP.1).

8.2.4.5 Argumentaire pour le support mutuel des exigences fonctionnelles

Comme démontré par l'argumentaire précédent, ainsi que par le respect des dépendances entre exigences fonctionnelles, les exigences fonctionnelles contribuent ensemble à satisfaire les objectifs de sécurité pour la TOE. De plus il n'existe aucun conflit entre les exigences fonctionnelles sélectionnée et définies dans cette cible de sécurité.

8.2.5. Argumentaire pour les mesures d'assurance

Le tableau ci-dessous justifie la nécessité des mesures d'assurance par rapport aux composants d'assurance Critères Communs sélectionnés. Il faut y ajouter MA.CRYPTO qui permet de satisfaire au processus de qualification de niveau standard défini par l'ANSSI dans [QUALIF_STD].

L'argumentaire détaillé est fourni au chapitre 6.2

Mesures d'assurance		MA.ENV_CONF	MA.ENV_SEC	MA.ENV_LIV	MA.ENV_SUP	MA.DEV	MA.TEST_DEV	MA.TEST_EVAL	MA.GUIDE_INST	MA.GUIDE_ADMIN	MA.GUIDE_UTILIS	MA.VUL
ADV_ARC.1	Description de l'architecture de sécurité					X						
ADV_FSP.3	Spécification fonctionnelle avec complete summary					X						
ADV_TDS.2	Conception « architectural »					X						
AGD_OPE.1	Operational user guidance									X	X	
AGD_PRE.1	Preparative procedures								X			
ALC_CMC.3	Autorisation controls	X										
ALC_CMS.3	Implementation representation CM coverage	X										
ALC_DEL.1	Procédures de livraison			X								
ALC_DVS.1	Identification des mesures de sécurité		X									
ALC_FLR.3	Systematic flaw remediation				X							
ALC_LCD.1	Developer defined life-cycle model	X										
ATE_COV.2	Analysis of coverage						X					
ATE_DPT.1	Testing: basic design						X					
ATE_FUN.1	Functional testing						X					
ATE_IND.2	Tests indépendants - par échantillonnage						X	X				
AVA_VAN.3	Focused vulnerability analysis											X

Tableau 9 : Couverture des exigences d'assurance sécurité par les mesures d'assurance

8.2.6. Pertinence du niveau d'assurance

Le niveau d'assurance EAL3 augmenté des composants ALC_FLR.3 et AVA_VAN.3 associé à une expertise de l'implémentation de la cryptographie a été choisi pour assurer la conformité au processus de qualification de niveau standard défini par l'ANSSI dans [QUALIF_STD]. Ce niveau d'assurance impose:

- Des tests indépendants effectués par l'évaluateur (l'utilisateur final est alors assuré que les fonctions de sécurité de la TOE sont implémentées comme spécifié)

- Une analyse de vulnérabilité indépendante effectuée par l'évaluateur (l'utilisateur final est alors assuré que la TOE est résistante à des attaques de pénétration effectuées par des attaquants possédant un faible potentiel d'attaque).

- L'évaluation de l'architecture de sécurité et de l'architecture logiciel incluant l'analyse de l'implémentation (fonctions cryptographiques seulement) pour vérifier qu'il n'y a pas de défaut de sécurité

- De bonnes pratiques en matière de développement (l'utilisateur final est alors assuré que le produit a été correctement et sécuritairement conçu et développé et que tous les éventuels défauts de sécurité ont été tracés, analysés et corrigé).

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 :

Exigences fonctionnelles de sécurité pour la TOE		F.CONTROLE_ACCES	F.ENTREE_SECURISEE	F.GESTION_DROITS	F.CONFIGURATION_TOE	F.GESTION_CLES	F.GESTION_CONTENEUR	F.OPERATIONS_CRYPTO
FCS_CKM.1	Génération de clés cryptographiques					X		
FCS_CKM.3	Accès aux clés cryptographiques					X		
FCS_CKM.4	Destruction de clés cryptographiques					X		
FCS_COP.1	Opération cryptographique	X	X			X		X
FDP_ACC.1	Contrôle d'accès partiel	X		X				
FDP_ACF.1	Contrôle d'accès basé sur les attributs de sécurité	X		X				
FDP_ITC.1	Importation depuis une zone hors du contrôle de la TSF		X					
FDP_RIP.1	Protection partielle des informations résiduelles					X		
FIA_UAU.1	Programmation de l'authentification	X	X	X			X	
FIA_UID.1	Programmation de l'identification	X	X	X			X	
FMT_MOF.1	Administration des fonctions de la TSF				X			
FMT_MSA.1	Gestion des attributs de sécurité			X				
FMT_MSA.2	Attributs de sécurité sûrs		X	X	X		X	
FMT_MSA.3	Initialisation statique d'attribut				X			
FMT_MTD.1	Gestion des données de la TSF				X			
FMT_SMF.1	Spécification des fonctions d'administration			X	X		X	
FMT_SMR.1	Rôles de sécurité	X			X			

Tableau 10 : Couverture des exigences fonctionnelles par les spécifications globales de la TOE

FCS_CKM.1 Génération de clés cryptographiques

A chaque conteneur est associé une clé de chiffrement et déchiffrement des fichiers. Cette clé est tirée lors de l'initialisation du conteneur (effectuée uniquement par l'Edition Standard). Elle répond aux critères de choix d'algorithme et de longueurs de clés configurées dans les policies. Par défaut, c'est une clé AES de 256 bits.

A chaque clé d'accès créée, une clé cryptographique est générée par la TOE.

La fonction de sécurité F.GESTION_CLES implémente cette exigence fonctionnelle.

FCS_CKM.3 Accès aux clés cryptographiques

L'accès aux clés cryptographiques gérées par la TOE est implémenté par la fonction de sécurité F.GESTION_CLES.

Cette fonction est utilisée lorsque la TOE :

- Récupère une clé d'accès pour accéder à la clé conteneur avant de pouvoir créer une nouvelle clé d'accès (création d'un nouvel accès),
- Récupère une clé d'accès avant de pouvoir utiliser la clé de conteneur et ouvrir le conteneur en lecture et écriture.

FCS_CKM.4 Destruction de clés cryptographiques

Lorsqu'un conteneur est supprimé, les clés cryptographiques relatives au conteneur sont détruites. De même lorsqu'un accès est supprimé (effectuée uniquement par l'Edition Standard), la clé d'accès correspondante est détruite.

La fonction de sécurité F.GESTION_CLES implémente cette exigence fonctionnelle.

FCS_COP.1 Opération cryptographique

La TOE effectue les opérations cryptographiques suivantes :

- Récupère une clé d'accès avant de pouvoir créer une clé de conteneur et chiffrer le conteneur,
- Récupère une clé d'accès pour déchiffrer la clé de conteneur avant de pouvoir créer une nouvelle clé d'accès,
- Récupère une clé d'accès avant de pouvoir déchiffrer la clé de conteneur, afin de pouvoir déchiffrer les fichiers du conteneur,

- Récupère un mot de passe afin d'en dériver une clé d'accès qui va chiffrer ou déchiffrer la clé de conteneur.
- Transmet la clé de conteneur chiffrée au porte-clés puis récupère la clé de conteneur déchiffrée par le porte-clés afin de pouvoir déchiffrer les fichiers du conteneur.

La fonction de sécurité F.OPERATIONS_CRYPTO, implémentent les opérations cryptographiques mises au service des autres fonctions.

Les fonctions F.GESTION_CLES (création de la clé d'accès) et F.CONTROLE_ACCES (vérification de la clé d'accès) utilisent les fonctions de dérivation des clés à partir des mots de passe.

La fonction F.ENTREE_SECURISEE utilise des fonctions de wrapping pour assurer le transfert sécurisé des clés entre la TOE et les porte-clés physique.

FDP_ACC.1 Contrôle d'accès partiel

Afin d'utiliser un conteneur géré par la TOE, l'utilisateur doit impérativement présenter une clé d'accès valide, associée au conteneur concerné. Cette exigence de sécurité est implémentée dans la TOE par les fonctions de sécurité

- F.GESTION_DROITS pour la configuration des accès au conteneur
- F.CONTROLE_ACCES pour le contrôle d'accès au conteneur

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

Afin d'utiliser un conteneur géré par la TOE, l'utilisateur doit présenter une clé d'accès valide, associée au conteneur concerné. Pour pouvoir mettre en place ce fonctionnement :

- des droits sont associés aux utilisateurs (F.GESTION_DROITS),
- et l'accès aux zones est donc contrôlé (F.CONTROLE_ACCES)

FDP_ITC.1 Importation depuis une zone hors du contrôle de la TSF

Des données nécessaires au bon fonctionnement de la TOE sont importées depuis l'extérieur de la TSF comme les clés d'accès ou les mots de passe saisis par l'utilisateur. Ce ne sont que des données, aucun attribut de sécurité n'est importé.

La fonction de sécurité F.ENTREE_SECURISEE implémente la communication de clés d'accès fournies en entrée vers la TOE, et couvre donc cette exigence.

FDP_RIP.1 Protection partielle des informations résiduelles

Cette exigence fonctionnelle est mise en œuvre par la fonction de sécurité F.GESTION_CLES qui gère l'effacement sécurisé des clés en mémoire.

FIA_UAU.1 Programmation de l'authentification

Aucune ouverture de conteneur n'est possible sans une phase préalable d'authentification et d'identification de l'utilisateur. Pour chaque authentification, les utilisateurs doivent présenter une clé d'accès valide. En revanche, la création, le renommage et la destruction d'un container, la destruction d'un fichier du conteneur sont possible sans authentification.

Cette exigence fonctionnelle est implémentée par :

- F.GESTION_DROITS pour la configuration des accès au conteneur
- F.CONTROLE_ACCES pour le contrôle d'accès au conteneur
- F.ENTREE_SECURISEE pour sécuriser la communication des données fournies en entrée vers la TOE.
- F.GESTION_CONTENEUR pour la gestion des opérations permises ou non sans authentification.

FIA_UID.1 Programmation de l'identification

Aucune ouverture de conteneur n'est possible sur la TOE sans une phase préalable d'authentification et d'identification de l'utilisateur. Pour chaque identification, les utilisateurs doivent présenter une clé d'accès valide. En revanche, la création, le renommage et la destruction d'un container, la destruction d'un fichier du conteneur sont possible sans identification.

Cette exigence fonctionnelle est implémentée par :

- F.GESTION_DROITS pour la configuration des accès au conteneur
- F.CONTROLE_ACCES pour le contrôle d'accès au conteneur
- F.ENTREE_SECURISEE pour sécuriser la communication des données fournies en entrée vers la TOE.
- F.GESTION_CONTENEUR pour la gestion des opérations permises ou non sans identification.

FMT_MOF.1 Administration des fonctions de la TSF

Seuls les administrateurs de la TOE peuvent déterminer le comportement, activer désactiver, modifier le comportement des fonctions de chiffrement des conteneurs.

Ils ont par exemple la possibilité de choisir l'algorithme de chiffrement des conteneurs lors de leur création (AES, 3DES, avec longueurs de clés de 128, 192 ou 256 bits) ou d'activer la génération de données d'audit.

La fonction de sécurité F_CONFIGURATION_TOE implémente cette exigence.

FMT_MSA.1 Gestion des attributs de sécurité

Seuls les administrateurs et les utilisateurs ont la possibilité de modifier ou supprimer l'attribut de sécurité « clé d'accès au conteneur » (Edition Standard seulement).

Cette exigence fonctionnelle est implémentée par F.GESTION_DROITS pour la configuration des accès au conteneur

FMT_MSA.2 Attributs de sécurité sûrs

Les fonctions de sécurité F.GESTION_DROITS, F.GESTION_CONTENEUR (initialisation du conteneur), F_CONFIGURATION_TOE (force des mots de passe) et F.ENTREE_SECURISEE (sécurité de la communication des attributs fournies en entrée vers la TOE) permet de garantir que l'unique attribut de sécurité « clé d'accès au conteneur» est sûr (clés d'accès gérées par l'Edition Standard seulement).

FMT_MSA.3 Initialisation statique d'attribut

La TSF permet aux administrateurs de la TOE de spécifier des valeurs initiales alternatives aux valeurs par défaut lorsqu'un objet ou une information est créé (force des mots de passe par exemple).

La fonction de sécurité F_CONFIGURATION_TOE met en œuvre cette exigence.

FMT_MTD.1 Administration des données de la TSF

Seuls les administrateurs ont la possibilité de gérer les stratégies de sécurité (ou « policies). Cette exigence est implémentée par la fonction de sécurité F_CONFIGURATION_TOE.

FMT_SMF.1 Spécification des fonctions d'administration

La TOE permet de réaliser :

- Les fonctions d'initialisation des paramètres utilisés par les fonctions de sécurité
- Les fonctions de gestion des conteneurs
- Les fonctions de gestion des clés et mots de passe (Edition Standard uniquement)
- La fonction de recouvrement (Edition Standard uniquement)

Cette exigence fonctionnelle est implémentée par les fonctions de sécurité :

- F_CONFIGURATION_TOE (configuration des policies)
- F.GESTION_CONTENEUR (gestions des conteneurs)
- F.GESTION_DROITS (gestion des clés et mots de passe, gestion de l'accès de recouvrement)

FMT_SMR.1 Rôles de sécurité

La TOE supporte les rôles utilisateur et administrateur.

Cette exigence est implémentée par F_CONFIGURATION_TOE qui fixe les droits administrateur pour le recouvrement et par F.CONTROLE_ACCES qui contrôle le droit d'accès pour le recouvrement.

8.4. Argumentaire pour les annonces de conformité à un PP

Cette cible de sécurité ne déclare aucune conformité à un Profil de Protection. Aucun argumentaire n'est donc requis.

9. Annexe A : Exigences fonctionnelles de sécurité de la TOE

Cette annexe contient les textes officiels de la partie 2 des Critères Communs en version 3.1 de septembre 2007 avec l'ensemble des opérations réalisées 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	Cryptographic key generation
FCS_CKM.3	Cryptographic key access
FCS_CKM.4	Cryptographic key destruction
FCS_COP.1	Cryptographic operation
FDP_ACC.1	Subset access control
FDP_ACF.1	Security attribute based access control
FDP_ITC.1	Import of user data without security attributes
FDP_RIP.1	Subset residual information protection
FIA_UAU.1	Timing of authentication
FIA_UID.1	Timing of identification
FMT_MOF.1	Management of security functions behaviour
FMT_MSA.1	Management of security attributes
FMT_MSA.2	Secure security attributes
FMT_MSA.3	Static attribute initialisation
FMT_MTD.1	Management of TSF data
FMT_SMF.1	Specification of Management Functions
FMT_SMR.1	Security roles

Tableau 11 : Exigences fonctionnelles de sécurité pour la TOE

9.1. Class FCS : Cryptographic support

FCS_CKM	**Cryptographic key management**
FCS_CKM.1	Cryptographic key generation
FCS_CKM.1.1	The TSF shall generate cryptographic keys in accordance with a specified cryptographic key generation algorithm *génération de nombres pseudo-aléatoires et diversification de clés* and specified cryptographic key sizes *clés symétriques de 128 à 256 bits* that meet the following: *PKCS#5 v2.0*. *Raffinement non éditorial :* *L'Edition limitée ne permet pas de générer de clé de chiffrement des fichiers (initialisation du conteneur).*
FCS_CKM.3	Cryptographic key access
FCS_CKM.3.1	The TSF shall perform *l'utilisation de clés* in accordance with a specified cryptographic key access method *déchiffrement (déwrapping) des clés par la clé d'accès* that meets the following: *Aucun*.
FCS_CKM.4	Cryptographic key destruction
FCS_CKM.4.1	The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method *réécriture de motifs composés de zéro* that meets the following: *Aucun*. *Raffinement non éditorial :* *L'Edition limitée ne permet pas de supprimer un accès et donc de détruire la clé associée.*
FCS_COP	**Cryptographic operation**
FCS_COP.1	Cryptographic operation
FCS_COP.1.1	The TSF shall perform *le hachage, le chiffrement, le déchiffrement, la génération de clés, le wrapping de clés et la dérivation de clés* in accordance with a specified cryptographic algorithm *SHA-256, RSA, 3DES et AES* and cryptographic key sizes *de 128 à 256 bits pour les clés symétriques et de 1536 à 2048 bits pour les clés asymétriques* that meet the following: *FIPS 180-2 (SHA-256), ANSI X9.52-1998 (3DES), FIPS 197 (AES) et PKCS#1 (RSA).* *Raffinement non éditorial :* *L'Edition limitée ne permet pas de générer de clé.*

9.2. Class FDP : User data protection

FDP_ACC	**Access control policy**
FDP_ACC.1	Subset access control
FDP_ACC.1.1	The TSF shall enforce the *SFP.ACCESS_OBJ on :* *Sujets: Utilisateurs de la TOE* *Objets: Conteneurs contenant les fichiers utilisateur et le fichier de contrôle.*
FDP_ACF	**Access control functions**
FDP_ACF.1	Security attribute based access control
FDP_ACF.1.1	The TSF shall enforce the *SFP.ACCESS_OBJ* to objects based on the following: *Sujets : Utilisateurs de la TOE* *Attributs de sécurité:Clés d'accès utilisateur permettant ou non d'ouvrir le conteneur.*
FDP_ACF.1.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *Objet : Conteneur* *Opération: Ouverture en lecture et écriture du conteneur* *Règle : authentification réussie auprès du conteneur à l'aide de la clé d'accès utilisateur.*
FDP_ACF.1.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: *Aucune*.
FDP_ACF.1.4	The TSF shall explicitly deny access of subjects to objects based on the *Aucune*.
FDP_ITC	**Import from outside TSF control**
FDP_ITC.1	Import of user data without security attributes
FDP_ITC.1.1	The TSF shall enforce the *SFP.ACCESS_OBJ* when importing user data, controlled under the SFP, from outside of the TOE.
FDP_ITC.1.2	The TSF shall ignore any security attributes associated with the user data when imported from outside the TOE.
FDP_ITC.1.3	The TSF shall enforce the following rules when importing user data controlled under the SFP from outside the TOE: *Aucune*.
FDP_RIP.1	**Residual information protection**
FDP_RIP.1	Subset residual information protection

FDP_RIP.1.1	The TSF shall ensure that any previous information content of a resource is made unavailable upon the *désallocation de la resource de* the following objects: *clés cryptographiques*.

9.3. Class FIA : Identification and authentication

FIA_UAU	**User authentication**
FIA_UAU.1	Timing of authentication
FIA_UAU.1.1	The TSF shall allow *la création, le renommage et la destruction d'un container, la destruction d'un fichier du conteneur* on behalf of the user to be performed before the user is authenticated.
FIA_UAU.1.2	The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UID	**User identification**
FIA_UID.1	Timing of identification
FIA_UID.1.1	The TSF shall allow *la création, le renommage et la destruction d'un container, la destruction d'un fichier du conteneur* on behalf of the user to be performed before the user is identified.
FIA_UID.1.2	The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

9.4. Class FMT : Security management

FMT_MOF	**Management of functions in TSF**
FMT_MOF.1	Management of security functions behaviour
FMT_MOF.1.1	The TSF shall restrict the ability to *déterminer le comportement, désactiver, activer ou modifier le comportement de* the functions *de chiffrement des conteneurs to administrateurs de la TOE*.

FMT_MSA	**Management of security attributes**
FMT_MSA.1	Management of security attributes
FMT_MSA.1.1	The TSF shall enforce the *SFP.ACCESS_ROLES* to restrict the ability to *modifier ou supprimer* the security attributes *accès au conteneur to utilisateurs et administrateurs.* *Raffinement non éditorial :* *Ce composant ne s'applique qu'à l'Edition Standard (l'Edition limitée ne permet pas de gérer les accès).*

FMT_MSA.2	Secure security attributes
FMT_MSA.2.1	The TSF shall ensure that only secure values are accepted for *les clés d'accès au conteneur*. *Raffinement non éditorial :* *Ce composant ne s'applique qu'à l'Edition Standard (l'Edition limitée ne permet pas de créer des accès).*
FMT_MSA.3	Static attribute initialisation
FMT_MSA.3.1	The TSF shall enforce the *SFP.ACCESS_ROLES* to provide *restrictive* default values for security attributes that are used to enforce the SFP.
FMT_MSA.3.2	The TSF shall allow the *administrateurs de la TOE* to specify alternative initial values to override the default values when an object or information is created. *Raffinement non éditorial :* *Ce composant ne s'applique qu'à l'Edition Standard (l'Edition limitée ne permet pas de gérer les accès).*
FMT_MTD	**Management of TSF data**
FMT_MTD.1	Management of TSF data
FMT_MTD.1.1	The TSF shall restrict the ability to *changer la valeur par défaut, interroger, modifier ou supprimer the stratégies de sécurité (policies)* to *administrateurs de la TOE.*
FMT_SMF	**Specification of Management Functions**
FMT_SMF.1	Specification of Management Functions
FMT_SMF.1.1	The TSF shall be capable of performing the following management functions: - *Les fonctions de gestion des accès* - *Les fonctions de gestion des conteneurs* - *La fonction de recouvrement* - *Les fonctions d'initialisation des paramètres utilisés par les fonctions de sécurité* *Raffinement non éditorial :* *L'Edition limité ne permet pas de gérer les accès (et donc d'affecter des clés de recouvrement).*
FMT_SMR	**Security management roles**
FMT_SMR.1	Security roles
FMT_SMR.1.1	The TSF shall maintain the roles *administrateur de la TOE et utilisateur de la TOE.*

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Copyright © Prim'X Technologies 2003, 2009.