Cible de sécurité CSPN

Forecomm — BlueFiles v4.16

- PUBLIC -

Version 1.0 - *Light*

Identification du document

Caractéristiques

Objet	Cible de sécurité CSPN - BlueFiles v4.16
Nombre de pages	23
Diffusion	PUBLIC

Historique

Version	Date	État
1.0	16/04/2021	Première version

Table des matières

1. Introduction

1.1. Identification du produit

Organisation éditrice	Forecomm
Lien vers l'éditeur	*https://www.forecomm.net/*
Nom du produit	BlueFiles
Numéro de la version évaluée	v4.16
Domaine technique CSPN	Communication sécurisée

1.2. Structure du document

Ce document est constitué, outre cette introduction, de 5 parties décrivant :

- la cible d'évaluation (TOE) ;
- l'environnement d'évaluation ;
- les biens sensibles à protéger par le produit ;
- les menaces qui pèsent sur ces biens sensibles ainsi que les agents menaçants identifiés ;
- les fonctions de sécurité couvrant les menaces précédemment décrites.

1.3. Références

[CEM]	Common Methodology for Information Technology Security Evaluation : Evaluation Methodology, version en vigueur.
[RGS_B]	Référentiel général de sécurité, annexes B : [RGS_B1] : Règles et recommandations concernant le choix et le dimensionnement des mécanismes cryptographiques. [RGS_B2] : Règles et recommandations concernant la gestion des clés utilisées dans des mécanismes cryptographiques. [RGS_B3] : Règles et recommandations concernant les mécanismes d'authentification.
[CSPN]	Certification de sécurité de premier niveau des produits des technologies de l'information, référence ANSSI-CSPN-CER-P-01, version en vigueur.
[CRITERES]	Critères pour l'évaluation en vue d'une certification de sécurité de premier niveau, référence ANSSI-CSPN-CER-I-02, version en vigueur.

2. Description du produit

2.1. Description générale du produit

BlueFiles est une solution logicielle permettant de réaliser des transferts de messages et fichiers de manière sécurisée, en appliquant un principe de chiffrement de bout en bout. De plus, la solution propose des fonctionnalités complémentaires à ces opérations de transferts de fichier, permettant notamment :

- de suivre l'accès par les destinataires aux fichiers transférés en temps réel (état de consultation, statistiques, etc.) ;
- d'ajouter, supprimer ou modifier les autorisations d'accès aux transferts réalisés ;
- aux destinataires de répondre à des transferts reçus, en sécurisant cette réponse, et les pièces qui peuvent y être jointes, ceci de la même manière que pour le transfert initial, c'est-à-dire par un chiffrement de bout en bout.

La solution offre la possibilité aux clients finaux d'administrer leurs comptes utilisateurs : les clients de la solution *BlueFiles* disposent d'un compte « Société », auquel est associé un nombre de licences leur permettant de créer des comptes utilisateurs rattachés à ce compte Société. L'administration de ces comptes permet de répartir les utilisateurs au sein d'équipes (chaque équipe pouvant alors disposer d'un paramétrage qui lui est propre), de leur attribuer des profils différents (administrateur / manager / simple utilisateur), etc.

2.2. Description de la manière d'utiliser le produit

Dans le cadre de cette évaluation, l'application web *BlueFiles* est utilisée. Celle-ci, installée *on-premise* et accessible via un navigateur, permet d'accéder aux différentes fonctionnalités proposées par la solution, notamment (selon le profil de l'utilisateur) :

- administration et paramétrage des comptes Société et Utilisateurs ;
- accès aux statistiques relatives à l'utilisation de la solution par les utilisateurs dépendant d'une société ;
- gestion des équipes au sein d'une société ;
- création et consultation de transferts ;
- accès aux statistiques relatives aux transferts effectués par l'utilisateur courant.

À noter que c'est la version Standard de l'application qui est considérée dans le cadre de l'évaluation. Plus spécifiquement, cette version permet la définition d'un unique compte Société pour l'instance *BlueFiles*, par opposition à la version Multi-Tenant qui permet d'en définir plusieurs.

Par ailleurs, un serveur *BluePass*, hébergé en mode *Saas*, géré par Forecomm, est également nécessaire à la fourniture et au stockage des clés de chiffrement et de signature des utilisateurs de *BlueFiles*. Le serveur *BluePass* est commun aux différentes instances *BlueFiles* déployées.

2.3. Description technique du produit

Le produit est architecturé dans un mode client-serveur. Les différentes fonctionnalités de l'application sont réparties entre deux serveurs : *BlueFiles* et *BluePass*.

2.3.1. Le serveur BlueFiles

Le serveur *BlueFiles* fait office d'autorité au sein de la solution. Ses rôles sont les suivants :

- intervention dans les mécanismes cryptographiques afin de fournir des éléments permettant le chiffrement / déchiffrement des fichiers transférés avec *BlueFiles* ;
- gestion des autorisations d'accès aux transferts *BlueFiles* ;
- vérification des autorisations d'accès aux transferts *BlueFiles*, en fonction des utilisateurs.

Seul ce composant figure dans le périmètre de l'évaluation CSPN.

2.3.2. Le serveur BluePass

Le serveur *BluePass* fait office d'autorité auprès des serveurs *BlueFiles*. Ses rôles sont les suivants :

- authentification de l'utilisateur ;
- stockage et fourniture de la clé privée (sous forme chiffrée) de l'utilisateur, qui sera utilisée par *BlueFiles* dans le traitement de déchiffrement des transferts ($k_{u\_privée\_chiffrement}$) ;
- stockage et fourniture de la clé privée (sous forme chiffrée) de l'utilisateur, qui sera utilisée pour signer les données, chiffrées, des transferts ($k_{u\_privée\_signature}$) ;
- stockage et fourniture des clés publiques des autres utilisateurs (destinataires des transferts), utilisées par *BlueFiles* dans le traitement de chiffrement des transferts ($k_{u\_publique\_chiffrement}$) ;
- stockage et fourniture des clés publiques des utilisateurs (émetteurs de transferts), permettant la vérification de la signature des données des transferts ($k_{u\_publique\_signature}$).

Comme indiqué ci-dessus, les clés privées des utilisateurs ne sont connues par le serveur *BluePass* que sous forme chiffrée : le chiffrement est réalisé, pour chacune de ces clés, à l'aide de clés symétriques dérivées du mot de passe de l'utilisateur.

Lors de l'authentification d'un utilisateur, ces clés sont déchiffrées localement (dans le navigateur internet de l'utilisateur) et peuvent alors être utilisées pour les traitements de signature et/ou déchiffrement qui les requièrent. Afin d'être accessibles à tout moment pendant la navigation sur le site web de la solution, ces clés privées sont stockées dans le *localStorage* du navigateur après avoir été chiffrées par des clés symétriques, $k_{session\_chiffrement}$ et $k_{session\_signature}$, dérivées via la fonction PBKDF2 utilisant une passphrase aléatoire dont une partie est présente sur le serveur et l'autre partie dans le *localStorage* du navigateur.

Une empreinte calculée à partir des clés publiques fournies par *BluePass* est consultable par l'utilisateur, qui pourra par la suite la fournir à ses contacts qui pourront l'utiliser pour confirmer l'identité de l'utilisateur lors de l'envoi ou de la réception de transferts par *BlueFiles*.

Cette empreinte est calculée par l'application de la fonction de hachage SHA256 sur le concaténé des clés publiques $k_{u\_publique\_chiffrement}$ et $k_{u\_publique\_signature}$. Elle est présentée à l'utilisateur lors de la création de ses bi-clés (ces jeux de clés sont créés localement dans le navigateur de l'utilisateur lors de la création de son compte), et est ensuite consultable à partir de son espace utilisateur sur le site de *BlueFiles*. Dans ce dernier cas, afin de garantir la validité des clés publiques fournies par *BluePass*, celles-ci sont validées contre les clés privées auxquelles elles se rapportent (déchiffrées par l'utilisateur lors de l'authentification). Si les clés publiques correspondent bien, l'empreinte est alors calculée et présentée à l'utilisateur dans les informations de son compte, il pourra alors la récupérer à cet endroit pour la communiquer à ses contacts.

À noter qu'il existe une unique instance du serveur *BluePass*, avec laquelle communiquent tous les serveurs *BlueFiles*.

2.3.3. Chiffrement d'un transfert

La création et l'émission d'un transfert sont décrites dans les sections suivantes. Pour rappel, comme indiqué dans la section Définition du périmètre de l'évaluation page 11, les restrictions suivantes sont appliquées dans le cadre de l'évaluation :

- Transfert classique : tous les destinataires disposent préalablement à la création du transfert d'un compte *BluePass*. À chaque compte sont associées les 2 bi-clés suivantes :

- $k_{u\_publique\_chiffrement}$ / $k_{u\_privée\_chiffrement}$ (stockée chiffrée)
- $k_{u\_publique\_signature}$ / $k_{u\_privée\_signature}$ (stockée chiffrée)

- Transfert par lien commun de partage : l'émetteur du transfert applique sur celui-ci une protection par mot de passe.

Le cas d'un dépôt de document via la page de dépôt d'un utilisateur se rapprochant fortement du cas d'un transfert classique (sans protection supplémentaire par mot de passe), il n'est pas décrit en détails dans cette section.

De plus, l'émetteur du transfert est supposé authentifié sur l'application.

2.3.3.1. Transfert classique

L'émission d'un transfert classique, dans le cas où tous les destinataires disposent préalablement d'un compte *BluePass*, consiste à réaliser le chiffrement des données via un algorithme AES-GCM, à l'aide d'une clé k_t de 256 bits générée aléatoirement. Cette clé k_t est ensuite elle-même chiffrée via les clés publiques des destinataires $k_{u\_publique\_chiffrement}$ par l'algorithme RSAES-OAEP.

L'émetteur a également la possibilité d'apposer un mot de passe sur son transfert : une clé k_{mdp_t} de 256 bits est alors dérivée du mot de passe, pour chiffrer k_t avant que celle-ci ne soit chiffrée par les clés publiques des destinataires ($k_{u\_publique\_chiffrement}$). Une empreinte du mot de passe est calculée, qui permettra sa vérification lors de l'accès au transfert, à l'aide de l'algorithme Scrypt (empreinte que nous dénommerons k_{mdp_scrypt}).

Une signature des données (après leur chiffrement par k_t) est réalisée avec la clé privée de l'émetteur $k_{u\_privée\_signature}$, ainsi qu'un calcul de HMAC-SHA256 (utilisant une clé k_{t_hmac}).

Toutes ces opérations sont réalisées localement, sur le poste de l'utilisateur. Seule les données chiffrées sont alors envoyées pour être stockées sur le serveur BlueFiles (données chiffrées du transfert, ainsi que les chiffrés des clés employées).

2.3.3.2. Transfert par lien commun de partage

L'émission d'un transfert par lien commun de partage protégé par mot de passe consiste à chiffrer les données via une clé k_t de 256 bits et l'algorithme AES-GCM. Une clé k_{mdp_t} est ensuite dérivée à partir d'un mot de passe choisi par l'émetteur. La clé k_t est alors chiffrée par la clé k_{mdp_t} par l'algorithme AES-GCM.

Comme pour le transfert classique :

- l'empreinte du mot de passe est calculée via l'algorithme Scrypt (pour obtenir k_{mdp_scrypt})
- une signature (par la clé $k_{u\_privée\_signature}$ de l'émetteur) est effectuée sur les données chiffrées,
- ainsi qu'un calcul de HMAC (par une clé k_{t_hmac}) sont également réalisés.

La différence avec le transfert classique réside dans le fait que seule k_{mdp_t} vient chiffrer k_t, celle-ci n'est en effet pas chiffrée par la clé $k_{u\_publique\_chffrement}$ des destinataires,

Toutes ces opérations sont réalisées localement, sur le poste de l'utilisateur. Seule les données chiffrées sont alors envoyées pour être stockées sur le serveur BlueFiles (données chiffrées du transfert, ainsi que les chiffrés des clés employées).

A la fin du traitement, un lien est communiqué à l'émetteur qu'il devra fournir aux contacts de son choix (ainsi que le mot de passe choisi pour protéger k_t).

2.3.4. Déchiffrement d'un transfert

L'accès au contenu d'un transfert est décrit dans les sections suivantes. Pour rappel, comme indiqué dans la section Définition du périmètre de l'évaluation page 11, seuls les cas suivants sont pris en compte dans le cadre de l'évaluation :

- transfert classique : tous les destinataires disposent préalablement à la création du transfert d'un compte *BluePass*.
- transfert par lien commun de partage : l'émetteur du transfert a appliqué sur celui-ci une protection par mot de passe.

De plus, le destinataire du transfert est supposé authentifié sur l'application.

2.3.4.1. Transfert classique

Pour accéder à un transfert classique, le destinataire doit dans un premier temps s'authentifier, ce qui lui permettra d'obtenir sa clé privée $k_{u\_privée\_chiffrement}$. Afin d'avoir accès à celle-ci, elle sera stockée sous une forme chiffrée dans le localstorage du navigateur (chiffrée par une clé $k_{session\_chiffrement}$).

Ainsi, lors de l'accès à un transfert classique, la clé k_t pourra être déchiffrée par la clé $k_{u\_privée\_chiffrement}$ (algorithme RSAES-OAEP).

Si un mot de passe est présent sur le transfert, l'utilisateur aura à le saisir au préalable, afin d'obtenir k_{mdp_t} (une vérification du mot de passe est au préalable effectuée, en application la fonction Scrypt dessus, et en vérifiant k_{mdp_scrypt} ainsi obtenu). Dans ce contexte, k_t, surchiffrée, pourra alors être déchiffrée par k_{mdp_t} (AES-GCM).

Une vérification de la signature sur les données chiffrées est également réalisée à l'aide de la clé $k_{u\_publique\_signature}$ de l'émetteur. Cette clé étant obtenue auprès de BluePass, l'utilisateur pourra vérifier sa légitimité en comparant son empreinte avec celle qui aura été fournie par l'émetteur au préalable.

Si toutes les étapes précédentes ont réussi, le déchiffrement, par la clé k_t (AES-GCM), des données peut alors s'effectuer. Il est à noter que toutes les opérations décrites sont réalisées localement, sur le poste de l'utilisateur.

Dans le cas où des fichiers au format PDF sont présents dans le transfert, une clé k_{t_pdf} sera utilisée pour chiffrer k_t avant de la stocker dans le localstorage du navigateur, permettant ainsi d'ouvrir les fichiers PDF dans d'autres onglets sans avoir à redemander le mot de passe à l'utilisateur lors de leur ouverture.

2.3.4.2. Transfert par lien commun de partage

Les opérations de déchiffrement des données transférées par lien commun suit exactement le même processus que pour les transferts classiques, à ceci-près que le destinataire n'a pas à s'authentifier, et qu'un mot de passe lui sera demandé, pour obtenir la clé k_{mdp_t}. La clé k_t est obtenue en la déchiffrant directement par k_{mdp_t} (AES-GCM).

La vérification du mot de passe (par obtention de k_{mdp_scrypt}) ainsi que la vérification de la signature sur les données chiffrées (par $k_{u\_publique\_signature}$ de l'émetteur) sont réalisées de la même manière.

Les données sont déchiffrées par k_t (AES-GCM) localement sur le poste de l'utilisateur.

2.4. Description de l'environnement prévu pour l’utilisation du produit

L'environnement d'utilisation est constitué de trois composants :

- Le navigateur web de l'utilisateur ;
- Le serveur *BlueFiles on-premise* ;
- Le serveur *BluePass* en mode *Saas*.

2.5. Description des dépendances

La solution *BlueFiles* est installée sur un système d’exploitation *Debian* (GNU Linux) dans une version 9.X (*Debian Stretch*).

Les dépendances suivantes sont nécessaires :

- Serveur web *Apache 2* (>= 2.4.X) avec activation du module *mod_rewrite* ;
- Base de données *MySQL* (5.5.46) ;
- *PHP*, version 7.3 avec les extensions suivantes :
  - *PDO* ;
  - *SimpleXML* ;
  - *Ctype* ;
  - *Curl* ;
  - *Date* ;
  - *Exif* ;
  - *Filter* ;
  - *Iconv* ;
  - *Intl* ;
  - *Json* ;
  - *Mbstring* ;
  - *MCrypt* ;
  - *OpenSSL* ;
  - *Pcre* ;
  - *Xml* ;
  - *Session* ;
  - *Libxml* ;
  - *Spl* ;
  - *Shmop* ;
  - *Xmlwriter*.
- Les bibliothèques *JavaScript* suivantes :
  - API *Web Cryptography* ;
  - *Forge* ;
  - *asmCrypto* ;
  - *Scrypt.js* ;

Le code de l'application *BlueFiles* étant protégé par la solution *ionCube*, il est également nécessaire d'installer le *loader* correspondant à la version de PHP utilisée, lors du déploiement de l'application.

Par ailleurs, certaines bibliothèques *PHP* sont nécessaires au fonctionnement de l'application *BlueFiles*. La plupart d'entre elles sont utilisées à la fois par les parties site et API de l'application :

- *nikic/fast-route* v1.3.0
- *pimple/pimple* v3.3.1
- *psr/container* 1.0.0
- *psr/http-message* 1.0.1
- *slim/slim* 3.12.3
-

De plus, la bibliothèque suivante est utilisée uniquement par la partie site : *slim/php-view* 3.1.0.

Enfin, les bibliothèques suivantes sont utilisées uniquement par l'API :

- *ezyang/htmlpurifier* dev-master
- *guzzlehttp/guzzle* 7.2.0
- *guzzlehttp/promises* 1.4.0
- *guzzlehttp/psr7* 1.7.0
- *paragonie/constant_time_encoding v2.4.0*
- *paragonie/random_compat v9.99.100*
- *phpmailer/phpmailer* v6.2.0
- *phpseclib/phpseclib* 3.0.5
- *phpseclib/phpseclib2_compat 1.0.1*
- *psr/http-client 1.0.1*
- *ralouphie/getallheaders* 3.0.3
- *robthree/twofactorauth 1.7.0*

2.6. Description des utilisateurs typiques concernés

Six profils d'utilisateurs existent :

- les utilisateurs classiques authentifiés utilisant le client web pour s'échanger des fichiers ;
- les managers d'équipes, pouvant gérer les membres de l'équipe et modifier certaines caractéristiques sur les documents échangés ;
- les administrateurs Société ayant accès à la gestion des utilisateurs et des équipes de la société ;
- les super administrateurs ayant accès à la gestion des administrateurs de la solution ;
- les utilisateurs non-authentifiés pouvant recevoir des fichiers via un lien commun de partage ;
- les utilisateurs non-authentifiés pouvant recevoir des fichiers et pour lesquels sont créées des clés temporaires. À noter que cette dernière catégorie d'utilisateurs n'est pas incluse dans le périmètre de l'évaluation CSPN, comme décrit dans la section Définition du périmètre de l'évaluation page 11.

2.7. Description des hypothèses sur l'environnement

H.1 H_ADMINISTRATION_SERVEUR_BLUEFILES

Le serveur sur lequel est déployé la solution *BlueFiles* est correctement administré, par des personnels non hostiles et compétents.

H.2 H_CONFIANCE_POSTE_CLIENT

L'appareil de l'utilisateur (et en particulier le navigateur utilisé pour accéder à l'application web *BlueFiles*) est bien administré, sain et de confiance.

Il est porté à l'attention du lecteur qu'aucune hypothèse n'est faite sur le composant *BluePass* sur laquelle s'appuie la solution *BlueFiles*. Ainsi, le serveur *BluePass* sera considéré comme potentiellement malveillant ou compromis dans le cadre de cette évaluation.

2.8. Définition du périmètre de l'évaluation

Les éléments suivants font partie du périmètre de l'évaluation :

- le chiffrement et le déchiffrement des transferts réalisés via l'application web *BlueFiles* ainsi que les mécanismes et éléments cryptographiques correspondants, réalisés selon les conditions supplémentaires décrites ci-dessous ;
- le dépôt de documents via la page de dépôt d'un utilisateur de l'application ;
- le stockage des transferts sur le serveur *BlueFiles* ;
- le stockage des clés de chiffrement des transferts sur le serveur *BlueFiles* ;
- la gestion des comptes utilisateur individuels et des droits associés au sein de l'application, dans les conditions décrites ci-dessous ;
- le mécanisme d'authentification par mot de passe des utilisateurs de la solution ;
- plus généralement, les différentes fonctionnalités proposées par l'application web *BlueFiles*, à l'exception de celles explicitement exclues ci-dessous.

En revanche, les éléments suivants **ne rentrent pas** dans le périmètre d'évaluation :

- le serveur *BluePass* (à l'exception des éléments stockés sur ce serveur et mentionnés ci-dessus) ;
- les éléments cryptographiques impliqués dans la signature des JWT échangés entre les applications *BlueFiles* et *BluePass* ;
- le mécanisme de compte utilisateur partagé ;
- le mécanisme d'authentification par certificat des utilisateurs de la solution ;
- l'utilisation de la solution *BlueFiles* via des moyens autres que l'application web, et notamment via les applications natives ou l'*addin* Outlook.

De plus, les conditions suivantes s'appliquent dans le cadre de cette évaluation :

- les journaux d'événements (*eventlogs*) sont activés sur l'application ;
- dans les cas de transferts classiques (c'est-à-dire pour lesquels la liste des destinataires est renseignée lors de la création du transfert), ne sera considéré que le cas où tous les utilisateurs impliqués disposent déjà d'un compte *BluePass*, et peuvent donc communiquer l'empreinte des clés publiques $k_{u\_publique\_chiffrement}$ et $k_{u\_publique\_signature}$ associées à leur compte *BluePass*, aux autres utilisateurs
- par ailleurs, pour les transferts classiques, on considérera que les utilisateurs de la solution vérifient systématiquement l'empreinte des clés publiques fournies par *BluePass* afin de vérifier qu'il s'agit bien de celles de leur(s) interlocuteur(s) supposé(s), ces dernières ayant été obtenues préalablement par un moyen tiers ;
- dans les cas de transferts par lien commun de partage (c'est-à-dire pour lesquels la liste des destinataires n'est pas connue lors de la création du transfert, et où il appartient à l'émetteur de communiquer les informations nécessaires à ses destinataires), ne sera considéré que le cas où une protection par mot de passe est appliquée sur le transfert.

3. Description de l'environnement technique dans lequel le produit doit fonctionner

3.1. Prérequis matériels

Un système en mesure d'installer les dépendances décrites dans la partie Description des dépendances page 9.

L'application *BlueFiles* sera utilisée dans un navigateur web choisi par l'utilisateur.

3.2. Système d'exploitation retenu

Le système *Debian* 9 *Stretch* est retenu pour la réalisation de l’évaluation.

3.3. Configurations type

L'application *BlueFiles* sera configurée avec l’options suivante : journaux applicatifs (*eventlogs*) activés.

4. Description des biens sensibles que le produit doit protéger

Les biens sensibles de la solution *BlueFiles*, compte tenu du périmètre à évaluer.

4.1. Biens métier

B.1 B_TRANSFERT

Un transfert *BlueFiles*, constitué d'un message et/ou d'un ou plusieurs fichiers, créé par un émetteur et à destination d'un ou plusieurs destinataires (ces destinataires pouvant être listés explicitement dans le cas d'un transfert classique, ou ne pas être explicités dans le cas d'un transfert par lien commun de partage).

Les transferts doivent être protégés en confidentialité, en intégrité et en authenticité.

B.2 B_CONFIGURATION

Les fichiers de configuration de l'application *BlueFiles* doivent être protégés en confidentialité et intégrité.

B.3 B_ADRESSES_MAIL_UTILISATEURS

Les adresses e-mail des utilisateurs, servant d'identifiant à la connexion.

Elles doivent être protégées en confidentialité vis-à-vis d'un utilisateur externe à l'application (dénoté par la mention « ext. » dans le tableau récapitulatif ci-dessous).

B.4 B_LISTE_TRANSFERTS_UTILISATEUR

La liste des transferts dont un utilisateur est l'émetteur ou l'un des destinataires.

Cette information doit être protégée en confidentialité et en intégrité.

B.5 B_JOURNAUX

Les journaux applicatifs de l'application.

Ils doivent être protégés en confidentialité et en intégrité.

	Confidentialité	Intégrité	Disponibilité	Authenticité
B_TRANSFERT	X	X		
B_CONFIGURATION	X	X		
B_ADRESSES_MAIL_UTILISATEURS	X (ext.)			
B_LISTE_TRANSFERTS_UTILISATEUR	X	X		
B_JOURNAUX	X	X		

4.2. Biens cryptographiques

B.6 B_SECRETS_UTILISATEUR

Les secrets d'authentification d'un utilisateur tels que son mot de passe.

Ils doivent être protégés en confidentialité et en intégrité lors de leur stockage sur le serveur *BluePass*, et lors de leur transit, sous leur forme initiale ou une forme dérivée, entre différents composants.

B.7 B_CLE_CHIFFREMENT_TRANSFERT

La clé de chiffrement k_t intervenant dans le chiffrement (par AES-GCM) des transferts réalisés.

Cette clé doit être protégée en confidentialité et en intégrité.

B.8 B_CLE_HMAC_TRANSFERT

La clé k_{t_hmac} intervenant dans le calcul de HMAC-SHA256 sur un transfert.

Cette clé doit être protégée en confidentialité et en intégrité.

B.9 B_CLES_PRIVEES_UTILISATEUR

La clé privée $k_{u\_privée\_chiffrement}$ associée au compte *BluePass* d'un utilisateur, intervenant dans le déchiffrement (par RSAES-OAEP) des clés k_t et k_{t_hmac}, ainsi que la clé privée $k_{u\_privée\_signature}$ utilisée pour la réalisation de la signature sur les données chiffrées.

Ces clés doivent être protégées en confidentialité et en intégrité lors de leur stockage sur le serveur *BluePass*, ainsi que lors de leur transit entre différents composants.

B.10 B_CLES_PUBLIQUES_UTILISATEUR

La clé publique $k_{u\_publique\_chiffrement}$ associée au compte *BluePass* d'un utilisateur, intervenant dans le chiffrement (par RSAES-OAEP) des clés k_t et k_{t_hmac}, ainsi que la clé publique $k_{u\_publique\_signature}$, qui permettra la vérification de la signature du transfert réalisée par l'émetteur de celui-ci.

Ces clés doivent être protégées en intégrité lors de leur stockage sur le serveur *BluePass*, ainsi que lors de leur transfert entre différents composants.

B.11 B_MOT_DE_PASSE_TRANSFERT

Le mot de passe appliqué à un transfert. Ce mot de passe n'est défini que pour les transferts par lien commun de partage, et pour les transferts classiques pour lesquels l'émetteur a choisi d'appliquer une protection supplémentaire par mot de passe.

Ce mot de passe doit être protégé en confidentialité.

B.12 B_CLES_SESSION_UTILISATEUR

Les clés $k_{session\_chiffrement}$ et $k_{session\_signature}$ générées à l'initiation de la session de l'utilisateur, utilisées pour le chiffrement respectif de ses clés privées $k_{u\_privée\_chiffrement}$ et $k_{u\_privée\_signature}$ (par AES-GCM), ces chiffrés étant stockés dans le *localStorage* du navigateur.

Ces clés doivent être protégées en confidentialité et en intégrité.

B.13 B_CERT_BLUEFILES

Le certificat TLS du serveur *BlueFiles* et la clé privée associée, ces deux éléments étant gérés par l'administrateur du serveur *BlueFiles*.

Ces éléments doivent être protégés en confidentialité et en intégrité.

	Confidentialité	Intégrité	Disponibilité	Authenticité
B_SECRETS_UTILISATEUR	X	X		
B_CLE_CHIFFREMENT_TRANSFERT	X	X		
B_CLE_HMAC_TRANSFERT	X	X		
B_CLES_PRIVEES_UTILISATEUR	X	X		
B_CLES_PUBLIQUES_UTILISATEUR		X		
B_MOT_DE_PASSE_TRANSFERT	X			
B_CLES_SESSION_UTILISATEUR	X	X		
B_CERT_BLUEFILES	X	X		

5. Description des menaces

5.1. Agents menaçants

AM.1 AM_RESEAU_BLUEFILES_CLIENT

Un attaquant en mesure d'intercepter les échanges entre un poste client et le serveur *BlueFiles* ou d'envoyer des requêtes vers celui-ci, sans nécessairement être utilisateur de la solution ou disposer d'un compte *BluePass*.

AM.2 AM_UTILISATEUR_MALVEILLANT

Un attaquant utilisateur de l'instance *BlueFiles* considérée (et possédant donc un compte *BluePass*).

AM.3 AM_BLUEPASS_MALVEILLANT

Un attaquant contrôlant le serveur *BluePass*, en mesure notamment d'accéder à des données arbitraires présentes sur celui-ci, et de les modifier.

AM.4 AM_RESEAU_BLUEPASS_BLUEFILES

Un attaquant en mesure d'intercepter les échanges entre les serveurs *BlueFiles* et *BluePass* ou d'envoyer des requêtes à l'un ou l'autre, sans nécessairement être utilisateur de la solution ou disposer d'un compte *BluePass*.

5.2. Menaces

M.1 M_ACCES_ILLEGITIME_TRANSFERT

Un attaquant parvient à accéder aux fichiers, chiffrés ou non, inclus dans un transfert.

M.2 M_ALTERATION_TRANSFERT

Un attaquant parvient à altérer le contenu d'un transfert lorsqu'il est stocké sur le serveur *BlueFiles*, ou lors de son envoi.

M.3 M_COMPROMISSION_SERVEUR_BLUEFILES

Un attaquant parvient à compromettre le serveur *BlueFiles* contenant notamment les clés de chiffrement des transferts, les vecteurs d'initialisation associés, les secrets des générateurs de code TOTP des comptes administrateurs et les fichiers de configuration de la solution.

M.4 M_COMPROMISSION_SERVEUR_BLUEPASS

Un attaquant parvient à compromettre le serveur *BluePass*. Il est alors en mesure de compromettre :

- les clé publiques ($k_{u\_publique\_chiffrement}$ et $k_{u\_publique\_signature}$) associées aux comptes des utilisateurs
- le service garant de l'authentification de l'utilisateur, lui permettant ainsi de pouvoir ouvrir une session auprès de *BlueFiles* au nom d'un autre utilisateur (victime).
    - Cependant, dans ce cas, il ne dispose pas pour autant des clés privées $k_{u\_privée\_chiffrement}$ et $k_{u\_privée\_signature}$, nécessaires au déchiffrement des transferts reçus, ainsi qu'à la signature des transferts émis, garantissant l'identité de l'émetteur de ceux-ci.

M.5 M_COMPROMISSION_COMPTE_UTILISATEUR

Un attaquant parvient à compromettre un compte utilisateur *BlueFiles* non privilégié ou un compte d'administrateur Société. Il est alors en mesure de déchiffrer tous les transferts passés (depuis le dernier renouvellement des bi-clés RSA de l'utilisateur) dont cet utilisateur est l'émetteur ou l'un des destinataires.

M.6 M_COMPROMISSION_ADMINISTRATEUR_SOLUTION

Un attaquant parvient à compromettre un compte administrateur de la solution *BlueFiles* et peut, si les bons droits sont positionnés, modifier le contenu des fichiers de configuration de l'application. Les droits en question ne sont pas donnés par défaut. Il est également en capacité de désactiver la journalisation des évènements et d'accéder à la clé HMAC utilisée pour vérifier l'intégrité des fichiers de journalisation.

M.7 M_ECOUTE_PASSIVE_FLUX

Un attaquant écoute les flux échangés entre un poste utilisateur et le serveur *BlueFiles* afin de récupérer les données sensibles en confidentialités échangées. Les informations circulant en clair entre le client et le serveur *BlueFiles* telles que les secrets d'authentification des utilisateurs sont alors compromis.

M.8 M_INTERCEPTION_FLUX

Un attaquant intercepte les flux générés entre un poste utilisateur et le serveur *BlueFiles*. Il les modifie dans le but d'altérer les données sensibles en intégrité.

Menaces planant sur les biens sensibles	
M_ACCES_ILLEGITIME_TRANSFERT	B_TRANSFERT B_CLE_CHIFFREMENT_TRANSFERT
M_ALTERATION_TRANSFERT	B_TRANSFERT
M_COMPROMISSION_SERVEUR_BLUEFILESq	B_TRANSFERT B_CONFIGURATION B_ADRESSES_MAIL_UTILISATEURS B_LISTE_TRANSFERTS_UTILISATEUR B_JOURNAUX B_CLE_CHIFFREMENT_TRANSFERT B_CLE_HMAC_TRANSFERT B_MOT_DE_PASSE_TRANSFERT B_CLE_SESSION_UTILISATEUR B_CERT_BLUEFILES
M_COMPROMISSION_SERVEUR_BLUEPASS	B_CLES_PUBLIQUES_UTILISATEUR
M_COMPROMISSION_COMPTE_UTILISATEUR	B_TRANSFERT B_ADRESSES_MAIL_UTILISATEURS B_LISTE_TRANSFERTS_UTILISATEUR B_SECRETS_UTILISATEUR B_CLES_PRIVEES_UTILISATEUR B_CLES_PUBLIQUES_UTILISATEUR B_MOT_DE_PASSE_TRANSFERT B_CLE_SESSION_UTILISATEUR
M_COMPROMISSION_ADMINISTRATEUR_SOLUTION	B_CONFIGURATION B_JOURNAUX
M_ECOUTE_PASSIVE_FLUX	B_TRANSFERT B_CONFIGURATION B_ADRESSES_MAIL_UTILISATEURS B_LISTE_TRANSFERTS_UTILISATEUR B_SECRETS_UTILISATEUR B_CLE_CHIFFREMENT_TRANSFERT B_CLE_HMAC_TRANSFERT B_MOT_DE_PASSE_TRANSFERT B_CLE_SESSION_UTILISATEUR
M_INTERCEPTION_FLUX	B_TRANSFERT B_CONFIGURATION B_ADRESSES_MAIL_UTILISATEURS B_LISTE_TRANSFERTS_UTILISATEUR B_SECRETS_UTILISATEUR B_CLE_CHIFFREMENT_TRANSFERT B_CLE_HMAC_TRANSFERT B_MOT_DE_PASSE_TRANSFERT B_CLES_SESSION_UTILISATEUR

	M.1	M.2	M.3	M.4	M.5	M.6	M.7	M.8
B.1	C	I, A	C, I, A		C		C	C, I
B.2			C, I			C, I	C	C, I
B.3			C, I				C	C, I
B.4			C, I		C, I		C	C, I
B.5			C, I			I		
B.6					C, I		C	C, I
B.7	C		C, I				C	C, I
B.8			C, I				C	C, I
B.9					C, I			
B.10				I	I			
B.11			C, I		C		C	C, I
B.12			C, I		C		C	C, I
B.13			C, I					

6. Description des fonctions de sécurité

FS.1 FS_CHIFFREMENT_TRANSFERTS

Un ensemble de mesures assure le chiffrement de bout en bout des transferts classiques, ainsi que des transferts par lien commun de partage protégés par mot de passe.

Ces mesures incluent la sécurisation des différents éléments cryptographiques mis en jeu lors de leur transit.

Outre leur confidentialité, l'intégrité des transferts est également garantie par les algorithmes utilisés.

FS.2 FS_STOCKAGE_CLES_SECRETS_SECURISE

Lorsqu'ils doivent être stockés, les différents secrets et clés manipulés au cours du processus de chiffrement/déchiffrement d'un transfert d'une part, et lors des processus de création de compte utilisateur et d'authentification d'autre part, le sont de manière sécurisée, plus précisément :

- k_t est stockée chiffrée (par RSAES-OAEP, en utilisant les clés publiques $k_{u\_publique\_chiffrement}$ des destinataires du transfert concerné dans le cas d'un transfert classique, ou bien, dans le cas d'un transfert par lien commun de partage protégé par mot de passe, par AES-GCM en utilisant la clé k_{mdp_t} d'une part et par RSAES-OAEP en utilisant la clé publique $k_{u\_publique\_chiffrement}$ de l'émetteur d'autre part) en base de données du serveur *BlueFiles*.

- k_{t_hmac} est stockée chiffrée (par RSAES-OAEP, en utilisant les clés publiques $k_{u\_publique\_chiffrement}$ des destinataires du transfert concerné dans le cas d'un transfert classique, ou bien, dans le cas d'un transfert par lien commun de partage protégé par mot de passe, par AES-GCM en utilisant la clé k_{mdp_t} d'une part et par RSAES-OAEP en utilisant la clé publique $k_{u\_publique\_chiffrement}$ de l'émetteur d'autre part) en base de données du serveur *BlueFiles*.

- $k_{u\_privée\_chiffrement}$ est stockée sous forme chiffrée (par AES-GCM, en utilisant la clé k_{u_mdp}, elle-même obtenue par dérivation via PBKDF2 du mot de passe de l'utilisateur) en base de données sur le serveur *BluePass*.

- $k_{u\_privée\_signature}$ est stockée sous forme chiffrée (par AES-GCM, en utilisant la clé k_{u_mdp}) en base de données sur le serveur *BluePass*.

- le mot de passe de l'utilisateur est stocké dérivé par *scrypt* sur le serveur *BluePass*.

- le mot de passe éventuellement appliqué au transfert est stocké chiffré par la clé du transfert k_t en base de données sur le serveur *BlueFiles*. Une dérivation par *scrypt* de ce mot de passe est également stockée dans cette même base de données.

À noter que les clés $k_{session\_chiffrement}$, $k_{session\_signature}$, $k_{u\_publique\_chiffrement}$ et $k_{u\_publique\_signature}$ ne sont pas couvertes par cette fonction de sécurité. Plus précisément, $k_{session\_chiffrement}$ et $k_{session\_signature}$ sont protégées en intégrité et en confidentialité par hypothèse (cf. **H_ADMINISTRATION_SERVEUR_BLUEFILES** et **H_CONFIANCE_POSTE_CLIENT** sur le serveur *BlueFiles* et le poste client respectivement). Les clés $k_{u\_publique\_chiffrement}$ et $k_{u\_publique\_signature}$ sont quant à elles couvertes en intégrité par **FS_INTEGRITE_CLES_UTILISATEUR**.

FS.3 FS_INTEGRITE_CLES_UTILISATEUR

Concernant les clés publiques $k_{u\_publique\_chiffrement}$ et $k_{u\_publique\_signature}$ des utilisateurs, stockées en clair sur le serveur *BluePass*, une compromission de ce dernier (correspondant à la réalisation de **M_COMPROMISSION_SERVEUR_BLUEPASS**) permet la remise en cause de l'intégrité de ces clés.

Pour répondre à cette problématique, les utilisateurs ont la possibilité de communiquer l'empreinte calculée à partir de leurs clés publiques aux autres utilisateurs, qui pourront alors manuellement vérifier l'intégrité de leurs clés $k_{u\_publique\_chiffrement}$ et $k_{u\_publique\_signature}$, fournies par le serveur *BluePass*.

Disposant des empreintes, les utilisateurs :

- lors de la création de transferts, peuvent s'assurer de l'intégrité des clés publiques, $k_{u\_publique\_chiffrement}$, des destinataires
- lors de la réception d'un transfert, peuvent s'assurer de l'intégrité de la clé $k_{u\_publique\_signature}$ de l'utilisateur émetteur, utilisée pour vérifier la signature des données chiffrées de ces transferts

Il est à noter qu'en cas de non vérification de ces empreintes, la compromission du serveur *BluePass* pourrait notamment résulter en une forme de déni de service sur la solution *BlueFiles*, en modifiant les clés publiques présentes en base, empêchant ainsi le chiffrement de nouveaux transferts par les clés publiques légitimes des utilisateurs, ce qui résultera en une erreur lors du déchiffrement de ces transferts (ou au déchiffrement en un "clair" incorrect).

Cependant, même dans ce cas, les besoins en confidentialité, intégrité et authenticité sur les transferts, et en confidentialité et intégrité sur les différentes clés listées ci-dessus, ne seraient pas remis en cause. En effet, la compromission du serveur *BluePass* ne permet que d'obtenir un chiffré des clés privées $k_{u\_privée\_chiffrement}$ et $k_{u\_privée\_signature}$ des utilisateurs, les clés symétriques utilisées pour réaliser le chiffrement n'étant pas stockées. Ainsi, la sécurité des biens métiers et des biens cryptographiques support ne se trouve pas menacée par une éventuelle compromission du serveur *BluePass*.

FS.4 FS_CLOISONNEMENT_CONTROLE_ACCES

L'application met en œuvre des mécanismes de cloisonnement et de contrôle d'accès garantissant qu'un utilisateur ne peut accéder qu'aux transferts (contenu et métadonnées) dont il est émetteur ou destinataire, ainsi qu'aux transferts par lien commun de partage protégé par mots de passe pour lesquels le lien et le mot de passe lui ont été transmis. Il ne peut ainsi pas lister les transferts adressés à ou émis par un tiers, ni accéder au contenu de ces transferts à moins qu'il l'en soit l'émetteur ou un autre destinataire.

FS.5 FS_ECHANGES_SECURISES

Les échanges entre le poste client et le serveur *BlueFiles*, ainsi qu'entre le serveur *BlueFiles* et le serveur *BluePass*, sont réalisés à travers des connexions sécurisées reposant sur le protocole HTTPS.

FS.6 FS_JOURNALISATION

Les événements ayant trait à l'authentification des utilisateurs, à la création de transferts et à l'accès à des transferts préalablement créés, sont journalisés sur le serveur *BlueFiles*.

FS.7 FS_AUTHENTIFICATION_SECURISEE_INTERFACE_ADMIN

L'accès aux fonctionnalités d'administration de l'instance *BlueFiles* est possible via une interface spécifique, pour les seuls super-administrateurs de cette instance, les comptes super-administrateurs étant spécifiques à une instance et décorrélés de la notion de compte *BluePass*. Les secrets d'authentification des super-administrateurs, en particulier leurs mots de passe, sont stockés de manière sécurisée sur le serveur *BlueFiles*. Les jetons de session sont quant à eux stockés de manière sécurisée, ont une durée de vie limitée et sont vérifiés lors de la réalisation d'une opération d'administration.

Couverture des menaces par les fonctions de sécurité et hypothèses	
FS_CHIFFREMENT_TRANSFERTS	M_ACCES_ILLÉGITIME_TRANSFERT M_ALTERATION_TRANSFERT
FS_STOCKAGE_CLES_SECRETS_SECURISE	M_COMPROMISSION_SERVEUR_BLUEFILES
FS_INTEGRITE_CLES_UTILISATEUR	M_COMPROMISSION_SERVEUR_BLUEPASS
FS_CLOISONNEMENT_CONTROLE_ACCES	M_ACCES_ILLÉGITIME_TRANSFERT M_ALTERATION_TRANSFERT M_COMPROMISSION_COMPTE_UTILISATEUR
FS_ECHANGES_SECURISES	M_ÉCOUTE_PASSIVE_FLUX M_INTERCEPTION_FLUX
FS_JOURNALISATION	M_COMPROMISSION_SERVEUR_BLUEFILES M_COMPROMISSION_COMPTE_UTILISATEUR M_COMPROMISSION_ADMINISTRATEUR_SOLUTION
FS_AUTHENTIFICATION_SECURISEE_INTERFACE_ADMIN	M_COMPROMISSION_ADMINISTRATEUR_SOLUTION
H_ADMINISTRATION_SERVEUR_BLUEFILES	M_COMPROMISSION_SERVEUR_BLUEFILES M_COMPROMISSION_ADMINISTRATEUR_SOLUTION
H_CONFIANCE_POSTE_CLIENT	M_COMPROMISSION_COMPTE_UTILISATEUR

	M.1	M.2	M.3	M.4	M.5	M.6	M.7	M.8
FS.1	X	X						
FS.2			X					
FS.3				X				
FS.4	X	X			X			
FS.5							X	X
FS.6			X		X	X		
FS.7						X		
H.1			X			X		
H.2					X			