Mise en place du déchiffrement OpenPGP

Table des matières

Objectif 2
Paramétrage Cryptographie (SCPS) 2
Notions clés 2
Cryptographie 2
OPENPGP 3
Paramétrage 3
Définition d'une clé de chiffrement de base de données 3
Configuration OpenPGP 4
Trousseau OpenPGP 4
Gestion du trousseau 4
Import de clés 6
Export de clés 7
Décodage manuel 7
Simulation chiffrement 8
Exploitation - mode automatique 9
Fichiers en entrée de signature 9
Fichiers en entrée du parapheur suivi de la signature 10
Fichiers en entrée du parapheur quelle que soit la finalisation 10
Exploitation - mode manuel 10
Notifications 10
Pas à pas 11
Ajout d'un fichier crypté en signature 11
Ajout d'un fichier crypté dans le parapheur (multi-tenants SXA) 12

Objectif

Certains ERP fournissent des fichiers bancaires chiffrés. Ces fichiers doivent alors être déchiffrés pour pouvoir être traités dans le parapheur ou dans la signature.

Paramétrage Cryptographie (SCPS)

Notions clés

Cryptographie

La cryptographie, c'est l'art de **protéger des informations** pour qu'elles ne soient lisibles que par les personnes autorisées. Elle repose sur des **mathématiques** et des **algorithmes** pour transformer un message clair en un message codé (chiffré).

Voici les notions clés, expliquées simplement :

1. **Chiffrement**
    a. On prend un message (ex. « Bonjour ») et on le transforme en une suite incompréhensible grâce à une **clé**.
    b. Pour le lire, il faut la **clé de déchiffrement**.
2. **Clés**
    a. **Symétrique** : la même clé sert à chiffrer et déchiffrer
    b. **Asymétrique** : deux clés différentes (publique pour chiffrer, privée pour déchiffrer).
3. **Objectifs**
    a. **Confidentialité** : seuls les destinataires lisent le message.
    b. **Authenticité** : vérifier que l'expéditeur est bien celui qu'il prétend être.
    c. **Intégrité** : s'assurer que le message n'a pas été modifié.

En pratique, ces opérations se font avec des algorithmes comme **RSA** ou **ECC**, et les clés sont très longues (2048 bits ou plus).

- **AES (Advanced Encryption Standard)** : très utilisé pour le chiffrement symétrique.
- **RSA** : basé sur la cryptographie asymétrique (clé publique/clé privée).
- **ECC (Elliptic Curve Cryptography)** : plus léger mais très sécurisé.

Exemple
Message original : Bonjour Blandine

- Etape 1 : chiffrement avec la clé publique

On utilise la clé publique du destinataire pour chiffrer.
Résultat (exemple simplifié) : U2FsdGVkX1+9a8kJ3h4zQ1rPz8vKj2Ml
Ce texte est illisible sans la clé privée.

- Etape 2 : déchiffrement avec la clé privée

Le destinataire utilise sa clé privée pour déchiffrer.
Résultat : Bonjour Blandine

OPENPGP

OpenPGP est **un standard ouvert de cryptographie** défini par l'**IETF** (Internet Engineering Task Force) dans la RFC 9580 .

Ce n'est **pas un logiciel**, mais **un format** qui décrit comment chiffrer, signer et échanger des données de manière sécurisée. Il est dérivé du logiciel historique **PGP (Pretty Good Privacy)** créé par Phil Zimmermann

OpenPGP repose sur la **cryptographie asymétrique** :

- Une **clé publique** pour chiffrer.
- Une **clé privée** pour déchiffrer.

Aspect	**PGP**	**GPG**
Licence	Propriétaire (à l'origine)	Libre (GPL)
Standard	Protocole PGP	Implémentation OpenPGP
Coût	Payant (souvent)	Gratuit
Compatibilité	Peut être limité	Large, interopérable

En résumé :

- **PGP** = le concept et le logiciel historique (souvent commercial).
- **GPG** = l'implémentation libre et moderne du standard **OpenPGP**.

Paramétrage

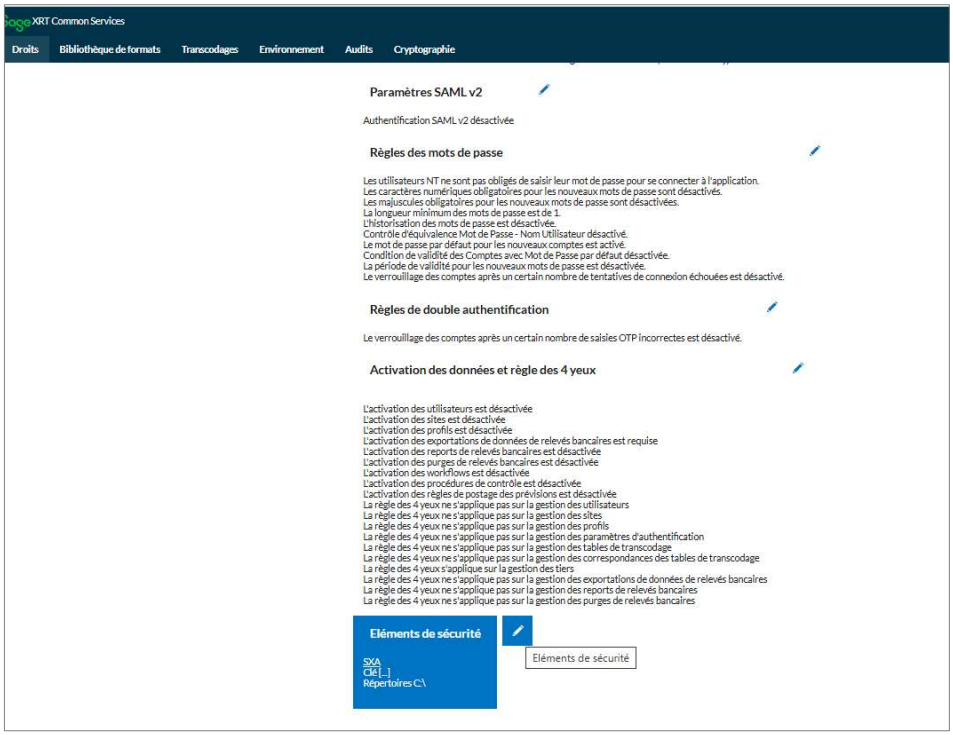
Définition d'une clé de chiffrement de base de données

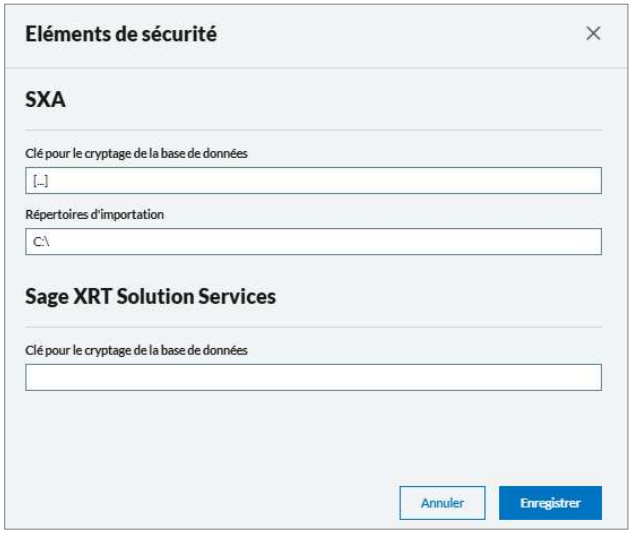
Etant donné le stockage d'informations confidentielles dans la base de données, il est nécessaire de définir une clé de cryptage de celle-ci afin de protéger ces données.

La clé de cryptage permet de **chiffrer** (rendre illisible) et **déchiffrer** (rendre lisible) les données. Dans le contexte d'une base de données, elle protège les informations sensibles contre tout accès non autorisé. Sans cette clé, les données restent incompréhensibles.

Une clé de cryptage peut être une **séquence de caractères générée par un algorithme cryptographique** ou une phrase complexe (appelée **passphrase**) qui sera transformée en clé binaire via un algorithme de dérivation (ex. PBKDF2, bcrypt, scrypt).
Cela permet à l'utilisateur de retenir une phrase plutôt qu'une suite complexe de caractères.

Dans le SCPS, à partir de la fonction Droits / Configuration de l'environnement, définir les clés de chiffrement des bases de données.





Cliquer sur Enregistrer pour sauvegarder la saisie.

Configuration OpenPGP

A partir du SCPS, utiliser la fonction **Configuration OpenPGP** sous le menu **Cryptographie**

Trousseau OpenPGP

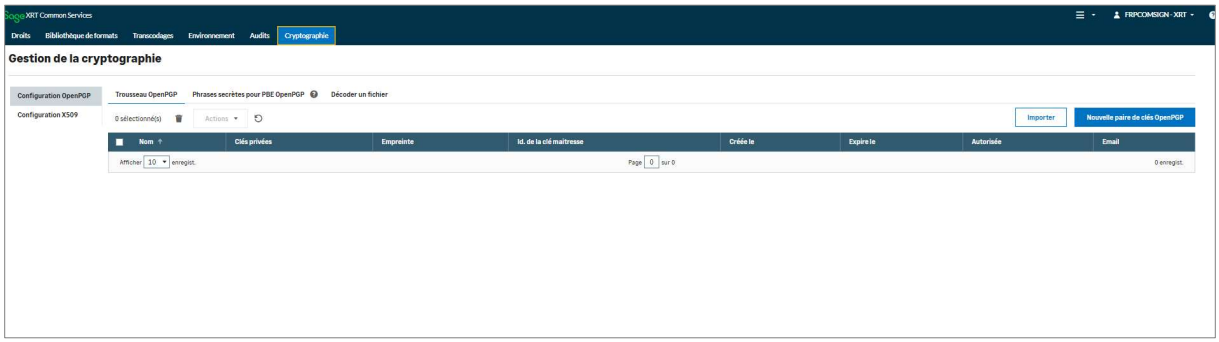
Gestion du trousseau

Un **trousseau de clés OpenPGP** (ou *keyring*) est un **fichier qui regroupe plusieurs clés OpenPGP** (publiques et/ou privées) ainsi que leurs certificats.

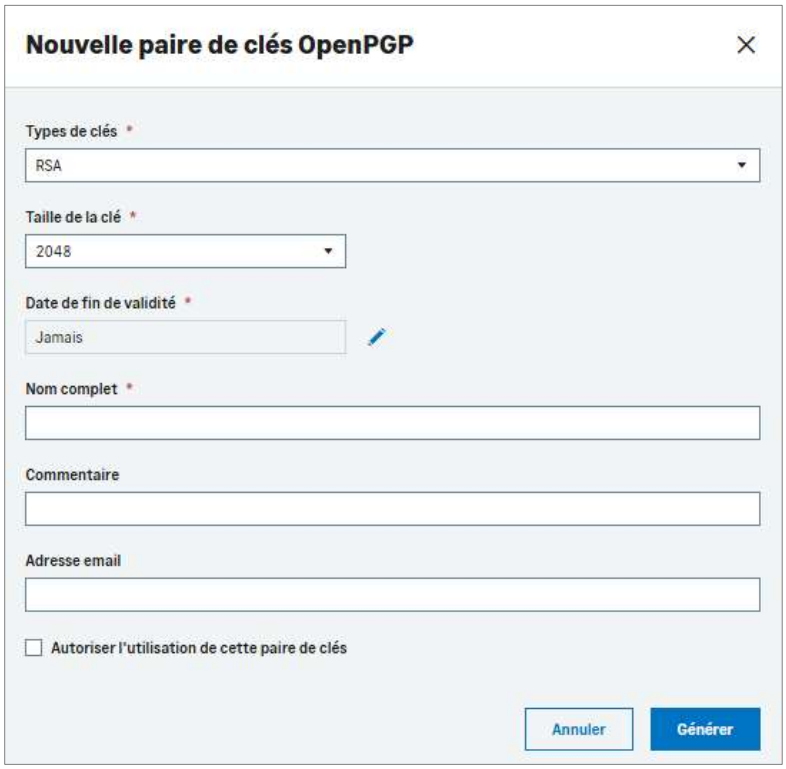
Il sert à **organiser et stocker** les clés nécessaires pour chiffrer, déchiffrer des messages ou des fichiers. Chaque clé peut contenir :

- Une **clé principale** (pour certifier et signer).
- Des **sous-clés** (pour chiffrer ou signer).

- Des **identités** (nom, email) liées à la clé.



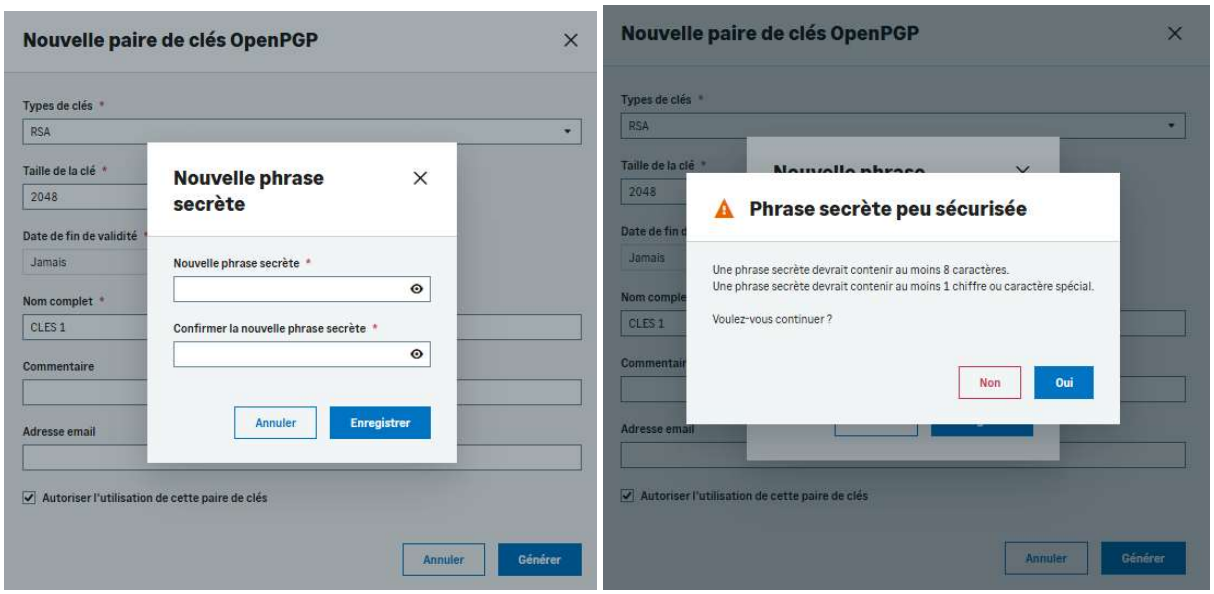
A partir de l'onglet **Trousseau OpenPGP**, utiliser le bouton **Nouvelle paire de clés OpenPGP**.



Renseigner obligatoirement le type de clés (RSA ou Curve), sa taille (de 1024 à 8192), sa date de fin de validité et un nom (attention les caractères accentués et particuliers ne sont pas acceptés).

Cocher la case Autoriser l'utilisation de cette paire de clés pour autoriser le déchiffrement avec la paire de clés en cours de création.

Cliquer sur le bouton Générer pour lancer la génération de la paire de clés.



Indiquer une phrase secrète (mot de passe) afin de finaliser la génération de la paire de clés. Si la phrase secrète est trop simple, un message demande la confirmation de cette phrase secrète en rappelant les caractéristiques d'une phrase sécurisée (selon la norme GPG). Cliquer sur Enregistrer pour sauvegarder la phrase secrète.

Si la clé de cryptage de la base de données n'est pas renseignée préalablement, la génération de la paire de clés ne pourra aboutir.

La paire de clés générée s'inscrit dans la liste des paires de clés composant le trousseau OpenPGP



La clé verte indique que cette paire de clés a une clé privée. En cas d'absence de clé privée, une clé rouge s'affiche.

L'empreinte de clé OpenPGP (ou *fingerprint*) est un identifiant unique et compact dérivé d'une clé publique OpenPGP grâce à une fonction de hachage cryptographique (ex. SHA-256). Elle sert à vérifier l'authenticité d'une clé sans avoir à comparer la clé entière, ce qui est très pratique car une clé publique peut être très longue. L'empreinte est utilisée pour identifier rapidement une clé dans un trousseau.

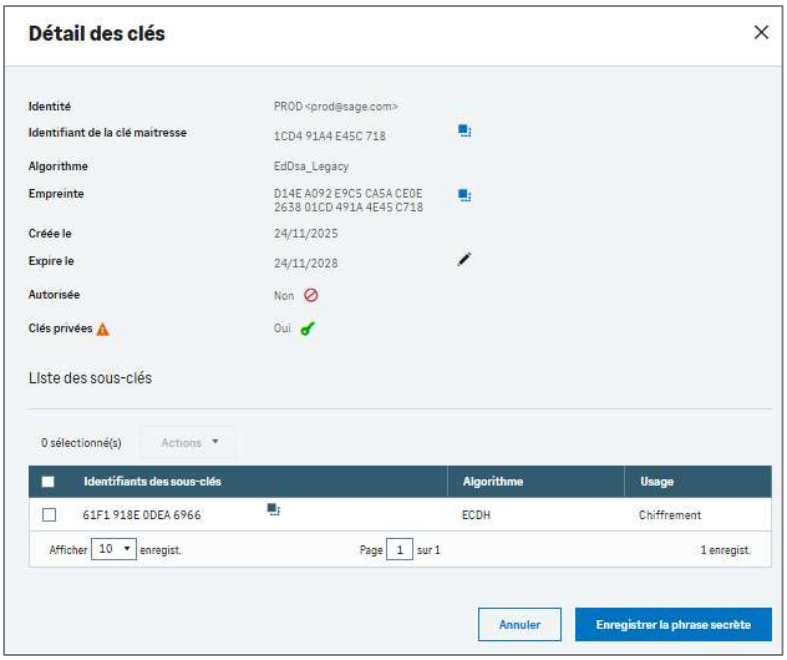
L'ID de la clé maîtresse dans une paire de clés OpenPGP est un identifiant unique qui permet de reconnaître la clé primaire (ou clé maîtresse) dans le trousseau. La clé maîtresse est la clé principale créée lors de la génération d'une paire de clés OpenPGP. Elle sert à certifier les sous-clés et les identités (nom, email). Elle est généralement utilisée pour la signature et la gestion des sous-clés.

Import de clés

Utiliser le bouton Importer pour sélectionner le fichier contenant la clé à importer. La clé importée n'est par défaut pas autorisée. L'enregistrement de la phrase secrète est requis.



Pour cela, utiliser le lien hypertexte sur le nom de la clé et utiliser le bouton Enregistrer la phrase secrète.



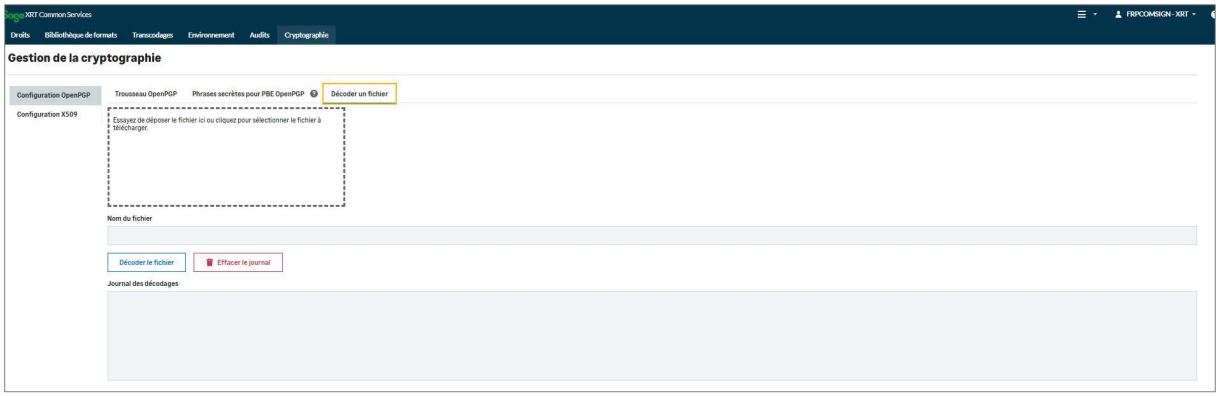
Export de clés

Il est également possible d'exporter les clés publiques et/ou les clés privées en sélectionnant la ou les paires de clé et en utilisant l'action Exporter les clés publiques ou Exporter les clés privées



Décodage manuel

A des fins de tests ou d'intervention rapide, il existe une fonction de décodage de fichiers chiffrés. Pour cela, utiliser l'onglet Décodage de fichiers. Sélectionner le fichier et utiliser le bouton Décoder le fichier.



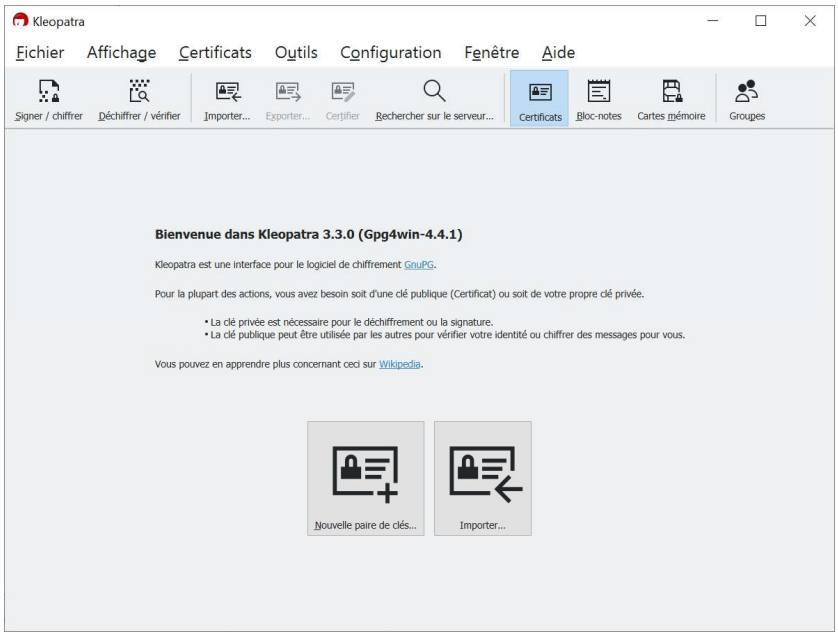
Simulation chiffrement

Pour simuler les cas de chiffrement, vous pouvez installer une application permettant de chiffrer :

gpg4win : https://gpg4win.org/download.html

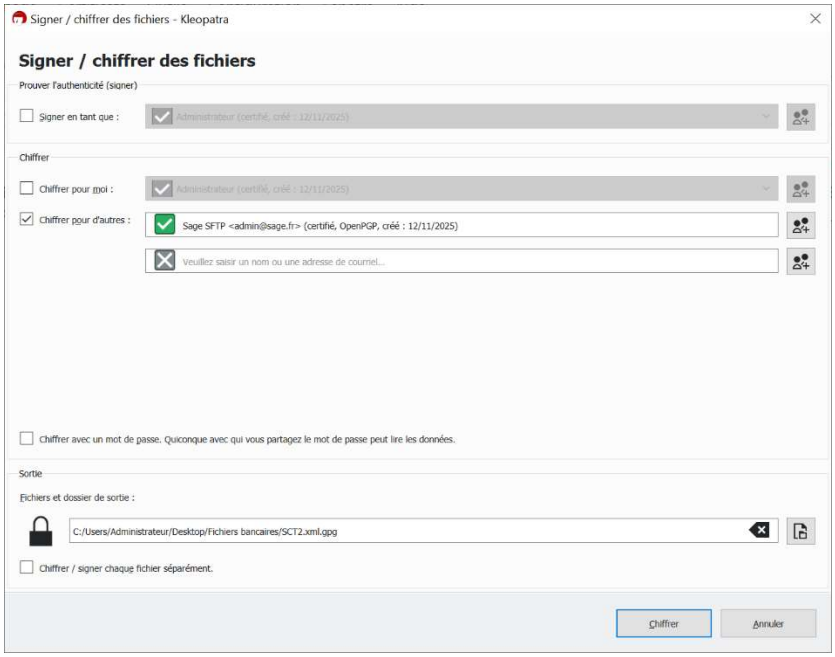


Gpg4win installe gpg.exe (GNU Privacy Guard), un outil en ligne de commande, ainsi que Kleopatra, une interface graphique, pour gérer la cryptographie OpenPGP.

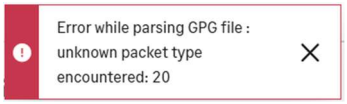


Il faut créer une nouvelle paire de clé et importer la clé privée, enregistrer la phrase secrète et autoriser son utilisation.

Pour chiffrer un fichier



Attention ! Lors du chiffrement, Kleopatra peut utiliser une extension AEAD (Authenticated Encryption with Associated Data) qui n'est actuellement pas supportée par le widget CRYPTO (BouncyCastle 2.6.2).



https://techychalkboard.com/2025/01/24/how-to-remove-the-aead-flag-in-gpg-and-verify-changes/

Exploitation - mode automatique

Fichiers en entrée de signature

Les 2 scripts existants ont été adaptés en ajoutant le paramètre /ENGINE:GPG pour permettre de déclarer si le fichier à ajouter dans la signature est crypté.

Exemple de la ligne de commande pour le script StdFrpPdsCopyFile.vbs (ajout d'un fichier crypté identifié directement en signature sans passer par le parapheur)

CScript StdFrpPdsCopyFile.vbs /REF /ROOTDIRECTORY:C:\temp\SFTP\FRPCOMSIGN /PARTNER:HPEBICS /CLIENT:SXACL01 /WORKGROUP:FRPCOMSIGN /ENGINE:GPG

Exemple de la ligne de commande pour le script StdFrpPdsCopy.vbs (ajout à partir d'un fichier crypté présent dans un répertoire sans passer par le parapheur)

CScript StdFrpPdsCopy.vbs /REF /ROOTDIRECTORY:C:\temp\SFTP\FRPCOMSIGN\REP /WORKGROUP:FRPCOMSIGN /ENGINE:GPG

NB : pour l'utilisation de ce script, la clé de cryptage de la base de données XRT Common Services doit être paramétrée sur le tenant COMSIGN.

Fichiers en entrée du parapheur suivi de la signature

Exemple de la ligne de commande pour le script StdFrpPdsCopyFile.vbs (ajout d'un fichier crypté identifié directement en parapheur)

CScript StdFrpPdsCopyFile.vbs /REF /ROOTDIRECTORY:C:\temp\SFTP\FRPCOMSIGN /PARTNER:HPEBICS /CLIENT:SXACL01 /WORKGROUP:FRPCOMSIGN /ENGINE:GPG /SEND:AF

Exemple de la ligne de commande pour le script StdFrpPdsCopy.vbs (ajout à partir d'un fichier crypté présent dans un répertoire dans le parapheur)

CScript StdFrpPdsCopy.vbs /REF /ROOTDIRECTORY:C:\temp\SFTP\FRPCOMSIGN\REP /WORKGROUP:FRPCOMSIGN /ENGINE:GPG /SEND:AF

NB : pour l'utilisation de ce script, la clé de cryptage de la base de données XRT Common Services doit être paramétrée sur le tenant COMSIGN.

Fichiers en entrée du parapheur quelle que soit la finalisation

Un nouveau script a été créé pour permettre de traiter un fichier crypté dans le parapheur indépendamment de sa finalisation (signature ou dépôt dans un répertoire).

Le pré-requis est que le nom du fichier débute par l'id du workflow suivi d'un _ (exemple : 48_fichier.txt où 48 est l'Id du workflow)

Exemple de la ligne de commande pour le script StdFrpParaphFile.vbs (ajout d'un fichier crypté identifié directement dans le parapheur)

CScript StdFrpParaphFile.vbs /ROOTDIRECTORY:C:\temp\SFTP\FRPCOMSIGN\PARAPH /WORKGROUP:SXACL01 /ENGINE:GPG

NB : pour l'utilisation de ce script, la clé de cryptage de la base de données XRT Common Services doit être paramétrée sur le tenant exploitant le parapheur.

Exploitation - mode manuel

Il est possible d'ajouter manuellement un fichier crypté en utilisant le widget d'ajout manuel.

Notifications

En cas d'erreur lors du décryptage, une notification peut être envoyée.

Pour cela, le paramétrage est à faire dans le fichier C:\ProgramData\XRT\XRTnOTIF.xml en ajoutant et en adaptant la section suivante :

```
<type t="CRYPTOGRAPHY" w="YOUR_WORKGROUP" from="xrtsolutions@sage.com">
<!--
```

Variable :
%NBRFILE% : number of file
Example of some status : 1 Failed to decode
Example :
<status s="1">
<to l="fr">p1.nom@sage.com; p2.nom@sage.com</to>
<to l="en">p3.nom@sage.com</to>
<to l="es">p4.nom@sage.com</to>
<subject>
<fr>Déchiffrement terminé avec erreur</fr>
<en>Decryption completed with error</en>
<es>Descifrado finalizado con error</es>
</subject>
<body>
<fr>Vous avez %NBRFILE% nouveau(x) fichier(s) en statut : Déchiffrement terminé avec erreur</fr>
<en>You have %NBRFILE% new file(s) with the status: Decryption completed with error</en>
<es>Tiene %NBRFILE% archivo(s) nuevo(s) con el estado: Descifrado finalizado con error</es>
</body>
</status>
-->
</type>

Pas à pas

Ajout d'un fichier crypté en signature

1. Sur le tenant COMSIGN, renseigner la clé de cryptage de la base de données XRT Common Services
2. Sur le tenant COMSIGN, créer une paire de clés OpenPGP (attention à bien cocher Autoriser l'utilisation de cette paire de clés)
3. Pour obtenir un fichier crypté, 2 possibilités
    a. avec une clé générée dans XRT Solution :
        i. exporter la clé publique (ex : fichier 13064D2764FFEDE5CE1810342BF4FC623BE1A0A3-BICLE1--pub.asc) depuis le widget.
        ii. importer la clé publique dans Kleopatra et chiffrer le fichier bancaire d'exemple avec. Un fichier avec une extension .gpg
    b. avec une clé générée dans un autre système :
        i. importer la clé privée (exemple fichier PROD_0x4E45C718_SECRET.asc) depuis le widget. Enregistrer la phrase secrète et autoriser son utilisation.
4. Se référer à la documentation pour respecter la nomenclature de nom du fichier en cas d'utilisation du script StdFrpPdsCopyFile.vbs ou la nomenclature du nom de répertoire en cas d'utilisation du script StdFrpPdsCopy.vbs (documentation « SXS 2026_Guide Ajout automatique de fichiers en signature ou dans le parapheur »)
5. Identifier le partenaire et le client concerné et paramétrer la ligne de commande comme dans l'exemple.
6. Lancer la ligne de commande et retrouver le fichier dans la liste des fichiers à signer.

Ajout d’un fichier crypté dans le parapheur (multi-tenants SXA)

Identique au cas précédent sauf que la gestion de la clé de cryptage de la base de données XRT Common Services et des paires de clé Open PGP se fait à partir du tenant SXA.