Réalisation d'une application graphique de hachage et crypto symétrique

Objectifs :

Il s'agit de réaliser un utilitaire graphique en mode local permettant de proposer :
- un ensemble de fonctions de hachage
- un outil de chiffrement AES
- et la transformation de fichiers (chiffrement/déchiffrement).

Le travail par équipe de deux est nécessaire pour arriver au bout des livrables. Une bonne coopération s'appuyant sur un partage équilibré des tâches favorisera le résultat.

Le fichier source à chiffrer « Jules Vern - Voyage au centre de la terre » est fourni avec ce sujet.

Scénario d'usage :

Un utilisateur opère une application graphique de chiffrement déchiffrement AES, avec calcul du hash incluant le salage.

Pour s'inspirer, on pourra se baser sur le logiciel QuickHash.

Plusieurs fonctionnalités s'offrent à l'utilisateur :

Fonctionnalités disponibles (à minima) selon l'arborescence du menu suivant :

```
1-Hacher un message

        -1.1 SHA-1
        -1.2 SHA-256
        -1.3 SHA-512
        -1.4 MD5
        -1.5 blake2b
```

l'option salage sera proposée :
par convention le « sel » sera un ID (256 bits par exemple) unique et aléatoire créé au démarrage de l'application. Il est associé à la concaténation des noms des membres du groupe assemblé par ordre alphabétique avec une représentation Camel case :

Nom évocateur	ID (256bits)
DupontDurandMartin1	ID1
DupontDurandMartin2	ID2
DupontDurandMartin3	ID3
DupontDurandMartin4	ID4

Ainsi pour l'équipe d'étudiants Martin, Dupont, on obtient pour les différents cas du scénario , 4 noms évocateurs symbolisant des destinataires potentiels:

```
2- Chiffrer/déchiffrer un message
        -2.1 gestionnaire de clés AES
                -2.1.1 créer/supprimer une clé AES (128-192-256)
                -2.1.2 activer désactiver une clé AES dans le catalogue (grisée et inexploitable)
                -2.1.3 gérer la liste/catalogue des clés avec un identifiant et un non évocateur
        -2.2 Chiffrer/déchiffrer un message avec AES
                -2.2.1 chiffrer avec une clé du catalogue/ rechercher/ouvrir un fichier sur le HDD
                -2.2.2 déchiffrer avec la clé correcte du catalogue/rechercher/ouvrir un fichier sur le HDD
```

Nom évocateur	ID (256bits)	Clé AES
DupontDurandMartin1	ID1	KAES1
DupontDurandMartin2	ID2	KAES2
DupontDurandMartin3	ID3	KAES3
DupontDurandMartin4	ID4	KAES4
.....	.....	.....

Soit finalement le tableau suivant servant de référence au chiffrement et hachage

---

Usage:

Dans l'usage, lors de la transformation d'un fichier source pour un groupe de destinataires potentiels, son nom évocateur sera donc sauvegardé dans un répertoire avec le fichier chiffré, ainsi que le hash généré associé à sa méthode de hash .

Dans un second temps, dans la phase de déchiffrement, cette organisation permettra de parcourir les répertoires de production et de retrouver la clé utilisée, réaliser le salage (récupération de ID associé au nom évocateur) pour finalement recalculer le hash sur fichier déchiffré et le comparer avec celui transmis.

En cas d'erreur un message est remonté à l'utilisateur (liste à définir) et mise en oeuvre d'actions correctives à déterminer (à justifier).

Pour chiffrer :

L'identifiant sera utilisé pour le salage selon les transformation suivantes (pseudo code).

```
chiffreAES (nomfichier1, KAES1) = fichier1chiffré
Hash (nomfichier1+ID1 + méthodes de hash) = hashfichier1
```

Pour déchiffrer (pseudo code):

```
dechiffreAES (fichier1chiffré, nom évocateur) = fichier1
Hash (fichier1, nom évocateur) )= hashcorrespondantOuiNon?
```

Organisation des répertoires :

Après avoir choisi un fichier à traiter, la méthode de hash ainsi que la clé AES du groupes de destinataires, les fichiers sont produits et gérés par répertoires :

```
/monrepertoiresource/
                monfichier1
                monfichier2
                monfichier3

/monrepertertoirechiffre/
                monfichier1/ fichier1chiffré, Méthode hash, hash correspondant au fichier1 non chiffré
(hashfichier1), le nom évocateur de la clé AES DurandDupontMartin1,
                monfichier2/ fichier2chiffré, Méthode hash, hash correspondant au fichier2 non chiffré
(hashfichier2), le nom évocateur de la clé AES DurandDupontMartin2,
                monfichier3/ fichier3chiffré, Méthode hash, hash correspondant au fichier3 non chiffré
(hashfichier3),le nom évocateur de la clé AES DurandDupontMartin3 ...

/monrepertoiredestinationdechiffre
                monfichier1/ fichier déchiffré sans le salage.
                monfichier2/ fichier déchiffré sans le salage.
                monfichier3/ fichier déchiffré sans le salage.
```

Au besoin vous pourrez rajouter d'autres paramètres dans les répertoires, sous réserve de ne jamais avoir simultanément les éléments secrets et en clair.

Technologies:

Langage disposant de librairies chiffrement/hachage et de gestion de fichiers en local.

- Python 3.8 avec par exemple les libs hashlib et pyca/cryptography est préconisé,
- à défaut NodeJs ou PHP.
- pour tout autre choix le préciser et le justifier .

Dans tous les cas, préciser et fournir son environnement permettant de l'exécuter, voire un exécutable si possible.

Equipes de réalisation :

Le travail sera réalisé par équipe de 2 personnes maximum.

Les équipes seront clairement identifiées dans l'entête du document rendu et sur les copies d'écran (à mettre dans une zone fixe de l'interface graphique)

Rendu :

- Dossier PDF avec
  - Fourniture d'un schéma l'architecture technique statique du logiciel
  - Copie d'écran des différents scénarios importants avec commentaires
  - Référence des bibliothèques (nom, version) et de leur composant utilisés (nom, version)
  - Résumé de l'organisation de l'équipe, des tâches réalisées par chacun et des points marquants du projet
- Fourniture du code commenté et son environnement permettant de l'exécuter
- Fourniture d'une archive avec les répertoires correspondant au fichiers utilisés/produits dans vos scénarios démontrant le fonctionnel de votre application (4 scénarios demandés)
- Oral de 15 minutes en fin de session de cours le 31 mars après midi

Les livrables seront remis Lundi 30/03/2020 soir à 23H30 dernier délai sur les adresses mail : francois.nicot@ynov.com