Projet Cryptographie

Auteurs :
DELAY Antoine
NARETTO Lilian
JAVAUD Corentin

Professeur Référent :
MAACHAOUI Mohamed

29 octobre 2021

Table des matières

1 Objectif

L'objectif de ce projet était de réaliser un procédé de diffusion électronique sécurisé de diplômes. Plusieurs parties sont nécessaires afin de le mettre en oeuvre :
- One Time Password
- Création du diplome
- Vérification du diplome

Etant familiarisé avec Flask, nous avons décidé de l'utiliser pour ce projet. Flask est un framework open source de développement web en Python. Il est très léger et a pour objectif de rester simple. C'est une technologie idéale pour ce projet car l'intérêt se porte plus sur le fond que la forme.

2 Guide d'utilisation

Afin d'utiliser Flask, de nombreux paquets Python sont nécessaires, les paquets nécessaires sont regroupés dans le fichier texte requirements.txt.
En utilisant la commande pip install et pip3 install on peut obtenir les modules.

On peut ensuite lancer l'application et le serveur avec python3 app.py.

Notre site web est simple mais efficace, on arrive en premier sur la page d'accueil.

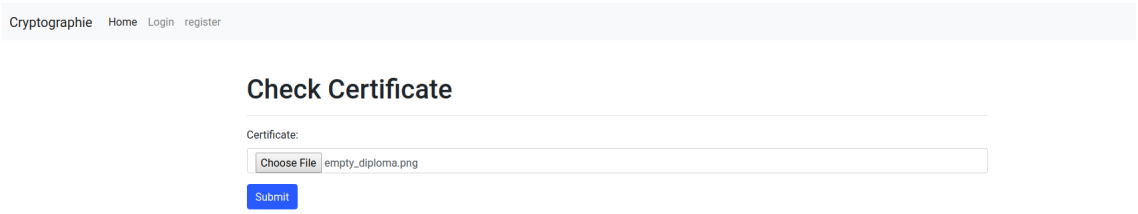

FIGURE 1 – Page d'accueil

Cette page permet de vérifier la validité des informations dissimulé par stéganographie dans le diplome et de vérifier la signature.

On a, en plus de la page d'accueil, un onglet de connexion et d'inscription.

FIGURE 2 – Page d'inscription

La page d'inscription demande 5 champs à l'utilisateur : Le prénom, le nom, l'email, le mot de passe et l'école. Il y a également un 6ème champ caché, c'est un booléen qui permet de savoir si l'utilisateur est un admin ou non. Nous avons créé a la main plusieurs admins. Il n'est pas possible de créer des admins à partir de cette page d'inscription.

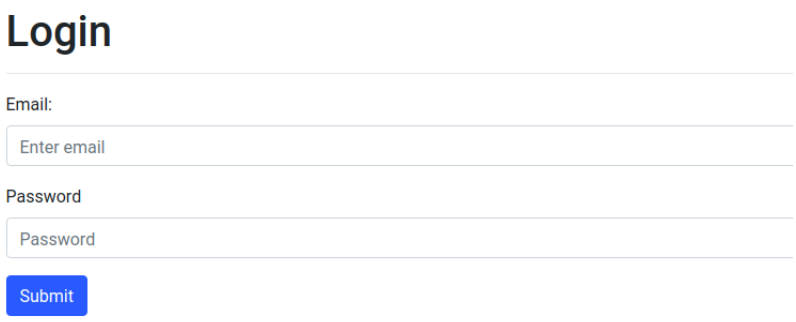

FIGURE 3 – Page de connexion

La page de connexion demande l'email et le mot de passe de l'utilisateur et va ensuite donner accès a d'autres onglets selon les privilèges de l'utilisateur.

Quand l'utilisateur n'est pas un admin, il arrive sur la page des diplomes. Elle permet aux élèves de faire une demande de création de diplome, cette demande est envoyée aux admins. Sur cette page les élèves peuvent également accéder aux diplomes validés, il est possible de les télécharger ou de les envoyer par mail.

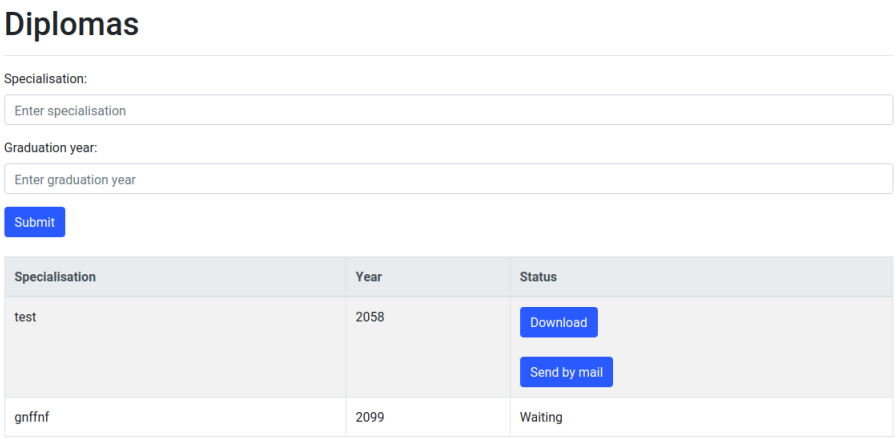

FIGURE 4 – Page des diplomes

Quand l'utilisateur est un admin, il a accès à la page d'admin avec ses fonctionnalités.

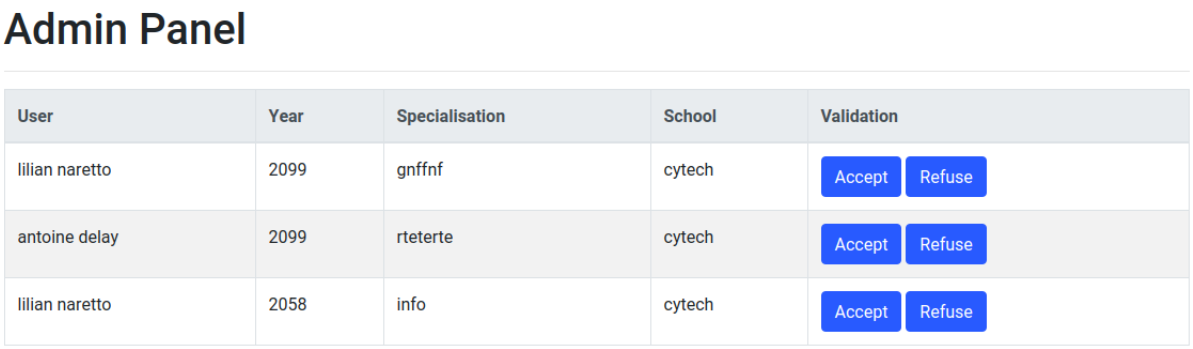

FIGURE 5 – Page d'admin

Sur cette page, l'admin peut refuser ou accepter une demande de création de diplome.

Pour accepter et donc creer le diplome, un OTP est demandé à l'admin. Le google authenticator est utilisé. Un QR Code est envoyé à l'adresse email de l'admin pour partager le secret et activer l'authenticator. L'admin peut enfin rentrer l'otp demandé et valider le diplôme.

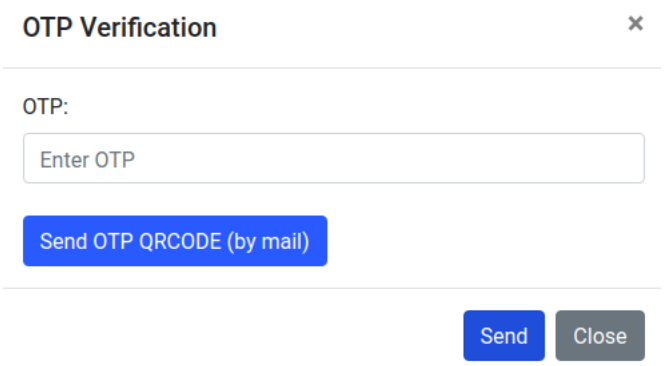

FIGURE 6 – OTP

3 Détail de fonctionnement

3.1 Schéma fonctionnel

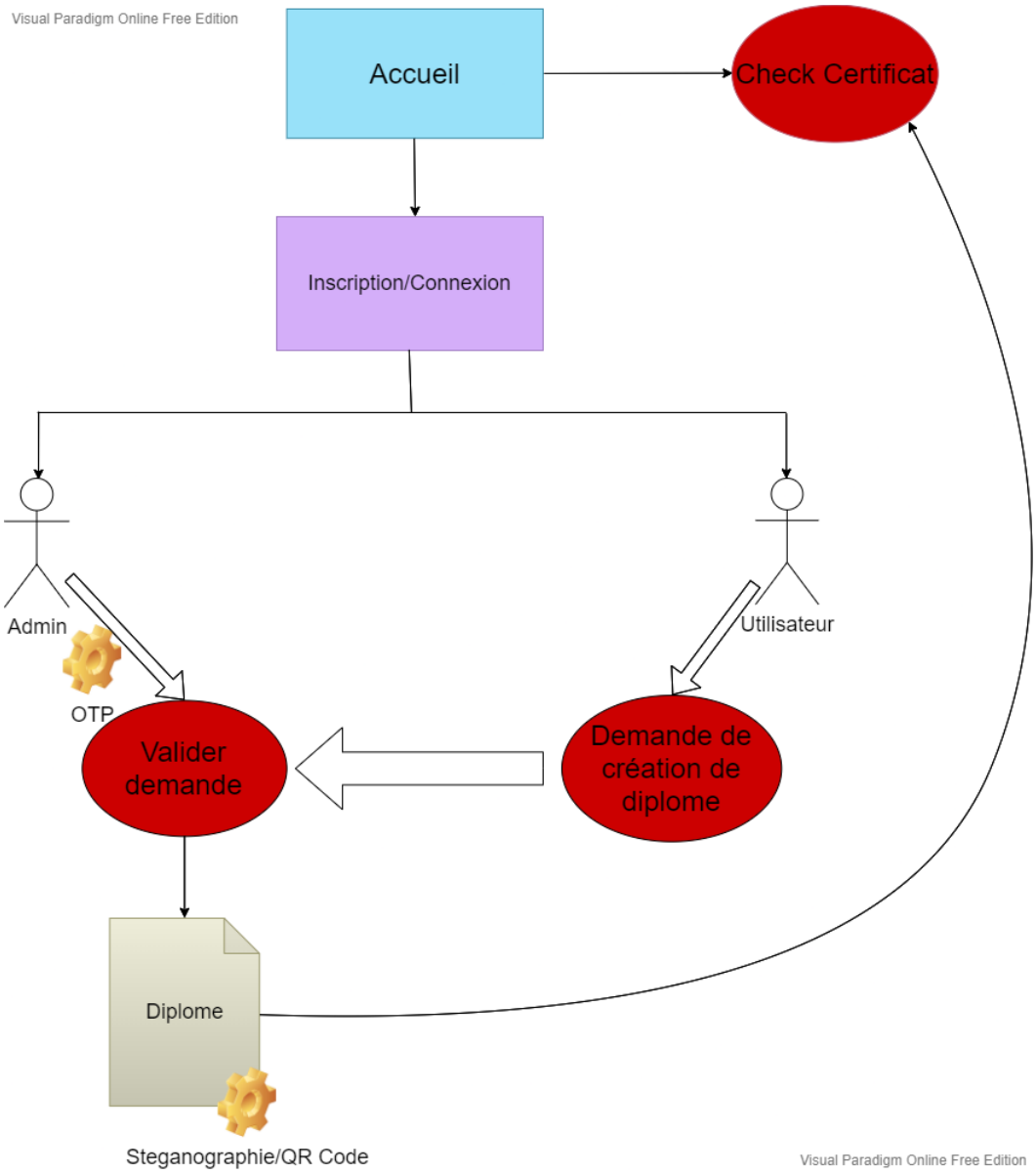

FIGURE 7 – Schema Fonctionnel

3.2 Publick Key Infrastructure

Une Publick Key Infrastructure (PK) est une hiérarchie de certificat. Le certificat root auto-signé va signer un autre certificat qui va lui même signer un autre certificat etc.

Une infrastructure de ce type a de nombreux avantage surtout lorsque différents services sont utilisés, chacun sur des serveurs et/ou autorités différents.

Dans le cadre du projet la mise en place d'une PKI n'a pas été necessaire. En effet l'avantage du PKI est de proposer une chaine de confiance entre différentes autorités, or on ne simule qu'un seul serveur l'intérêt est moindre. De plus pour vérifier le certificat qui sert à signer nos données, il sera nécessaire de remonter toute la chaine de certificat jusqu'au certificat root et de tous les verifier, ce qui peut être couteux en calcul.

4 Détails Du Code

4.1 La base du site en flask

Les fichiers majeurs à la création du site web sous flask sont app.py et manager.py. manager.py gère tout ce qui est base de données, il crée les bases et fait des requêtes sql. app.py gère tout ce qui est requêtes de POST sur le site et également qui peut voir quoi sur le site (distinction entre page admin et page utilisateur).

app.py appelle à lui plusieurs fonctions situées dans différents fichiers, comme totp.py et cryptfunction.py . totp.py est le fichier regroupant les fonctions d'envoi d'otp, de verificaiton d'otp et d'envoi du diplôme par mail, cryptfunction gère quant à lui la création des diplômes (envoyés dans le dossier Diplomas) et la verification (DLDiplomas).

Pour résumer la construction sous flask, nous sommes ici sous un format MVT, les vues renvoient des templates avec des informations, ces vues gèrent également les requêtes ainsi que l'accès aux pages, le site est entièrement sécurisé, un user ne pourra pas voyager dans les pages admin ou exécuter des fonctions.

4.2 Générer le Diplome

4.2.1 Générer le visuel

La première étape pour creer un diplome unique c'est d'écrire ce qui le caractérise et ce qui caractérise le diplomé. Pour nous aider, on utilise la bibliothèque python PIL et en particulier les classes Image et ImagesDraw. Image va nous servir a ouvrir les fichiers images dans python, ImageDraw a ecrire les informations liées au diplomé au pixel de l'image ou l'on souhaite l'écrire (cf code)

4.2.2 Générer le certificat

Afin de signer le diplôme nouvellement généré il nous faut une autorité représentant CY-Université. Pour certifier son authenticité on génère un certificat auto-signé avec openssl :

openssl req -x509 –config root-ca-cert.cnf -newkey rsa :2048 –out ca.pem –keyout private/ca.key –days 1826

Cette ligne de commande génère un certificat pour la clé publique de la pair clé privé/publique, générer par l'algorithme RSA 2048.

Une fois les données liées à l'établissement rentrées, le certificat est généré.

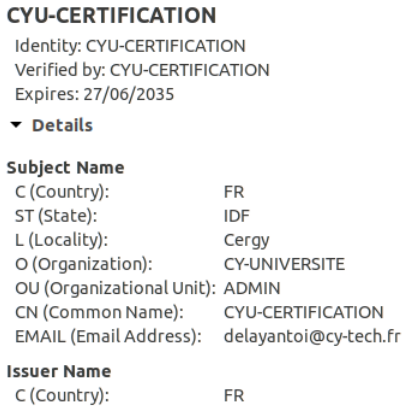

FIGURE 8 – certificat

4.2.3 Générer le Qrcode et la stéganographie

Afin de rendre le diplome plus difficile à falsifier, on utilise 2 méthodes :

Tout d'abord, on "cache" les données liées à l'utilisateur, en encodant une chaine de caractère le représentant dans les bits faibles de l'image, c'est la stéganographie.

Cette même chaine de caractère est transmise à la fonction sign qui à l'aide la clé publique du certificat de Cy-universite, va être encrypté.

Le token récuperé est encodé dans le QRCode lui même étant placé sur le diplôme.Le decodage du QRCode se fait donc sur l'application avec la clé privé de CY-Université

On a utilisé la clé publique pour encrypter les données en pensant que celle-ci devait être lu uniquement par l'application que l'on mette a disposition. Aussi ce que l'on a fait s'apparente plus à du chiffrement qu'a une signature.

4.3 OTP

Pour valider une demande de diplôme, l'admin doit utiliser un One Time Password. Cet OTP expire rapidement et ne peut pas être réutilisé. Cela permet d'ajouter une couche de sécurité à la création de diplôme et à s'assurer que l'admin est bien celui qui valide. Un secret est partagé entre l'admin et le serveur.Il existe plusieurs types d'OTP comme le Time Based OTP ou bien le Counter based OTP.

Nous avons décidé utiliser le Time Based en s'aidant de la librairie PyOTP. Celui-ci se base sur le temps et le secret pour générer l'otp. Pour partager le secret on utilise un QR Code envoyé par mail. Le Google authenticator est utilisé pour lire le QRCode et le TOTP peut être généré.