Gestionnaire de mots de passe décentralisé

Rapport d'avancement PX511

Sommaire

I) Introduction

Avec un nombre croissant de comptes à créer sur Internet pour pouvoir avoir accès à tel ou tel service, se pose la problématique des mots de passe : respecter les bonnes pratiques recommandées par l'ANSSI[1] pour trois ou quatre mots de passe est faisable, mais d'après NordPass[2], une personne moyenne aurait environ 70 mots de passe actifs. Nous sommes alors tous tentés de réutiliser les mêmes mots de passe pour plusieurs comptes : d'après une enquête d'Avast[3] en 2019, 93% des français ne respectent pas les bonnes pratiques en termes de mots de passe. Une solution a émergé pour répondre à ce problème : les gestionnaires de mots de passe. À partir d'un seul mot de passe maître (qui doit, celui-ci, être parfaitement sécurisé), il est possible d'accéder à tous ses mots de passe sans les retenir, mais aussi de créer des mots de passe sécurisés. Plus besoin de créer ses mots de passe, plus besoin de les retenir, tout cela de manière bien plus sécurisée qu'avant, les gestionnaires semblent la solution idéale.

Aujourd'hui, les gestionnaires de mot de passe connaissent un essor certain : il y a une prise de conscience générale de leur apport en termes de sécurité. Néanmoins, leur utilisation reste parfois obscure au grand public, les solutions sont souvent payantes car

[1][1]https://www.ssi.gouv.fr/guide/mot-de-passe/
[2]https://www.newswire.com/news/new-research-most-people-have-70-80-passwords-21103705
[3]https://press.avast.com/fr-fr/enqu%C3%AAte-avast-93-des-fran%C3%A7ais-utilisent-des-mots-de-passe-faibles

elles passent par un tiers de confiance, ou ne présentent pas de synchronisation des mots de passe entre différents appareils.

II) Définition des objectifs de l'étude théorique

L'objectif final est de trouver une solution gérant les mots de passe de manière simple (accessible au plus grand nombre), décentralisée, et assurant la synchronisation.
Précisons ce que nous entendons par "synchronisation" et "décentralisation".

Étant donnée une liste de mots de passe stockée sur un de nos appareils, nous devons pouvoir les partager à d'autres appareils (du même utilisateur) afin qu'ils puissent également les stocker et que nous puissions accéder à nos mots de passe depuis ces autres appareils. Ainsi, si nous créons un mot de passe sur un appareil, nous pourrons mettre à jour la liste de mots de passe sur un deuxième appareil afin de pouvoir utiliser ce nouveau mot de passe : c'est la synchronisation.

De plus, nous souhaitons éviter la centralisation des mots de passe, c'est-à-dire que pour accéder aux mots de passe, les appareils ne devront pas interroger une unique entité qui leur délivrera le mot de passe demandé (traditionnel modèle client-serveur). Cela pour éviter le coût que peut représenter un serveur, qui rendrait la solution payante et donc restreindrait l'accès au grand public.

Ainsi, nous ferons en premier lieu une brève étude du fonctionnement général d'un gestionnaire de mots de passe actuel, afin d'en dégager les problématiques et de mieux appréhender leurs fonctionnalités.

Ensuite, nous établirons une liste exhaustive des solutions de communication entre deux appareils (pour partager les mots de passe), nous définirons les avantages et les défauts de chacune de ces solutions afin de les comparer et de dégager celles qui semblent le plus adaptées à notre solution. Nous nous attarderons sur leur capacité à effectuer la synchronisation de la manière la plus efficace possible, ainsi qu'à respecter la décentralisation.

III) État de l'art

1) Fonctionnement d'un gestionnaire

De manière générale, un gestionnaire de mots de passe fonctionne de la manière suivante : un mot de passe maître est créé en premier. Ensuite, pour s'authentifier auprès du serveur (afin de créer un mot de passe ou bien d'accéder à un de vos services), il faudra saisir ce mot de passe maître. Tous vos mots de passe sont stockés au même endroit (dans la base de données du serveur). Quand vous voulez accéder à vos mots de passe, vous saisissez le mot de passe maître, qui est ensuite dérivé deux fois de manière différente, une fois pour obtenir la clef qui servira à déverrouiller le coffre-fort contenant vos mots de passe (cette clef n'est jamais envoyée au serveur, il n'y a que vous qui la possédez), et une autre fois pour obtenir la clef qui permettra de vous authentifier auprès du serveur, qui vous délivrera alors le coffre-fort contenant vos mots de passe. Ainsi, même si le coffre-fort est

intercepté, il ne pourra être déverrouillé car seul vous possédez la clef, qui n'est jamais envoyée sur le réseau.

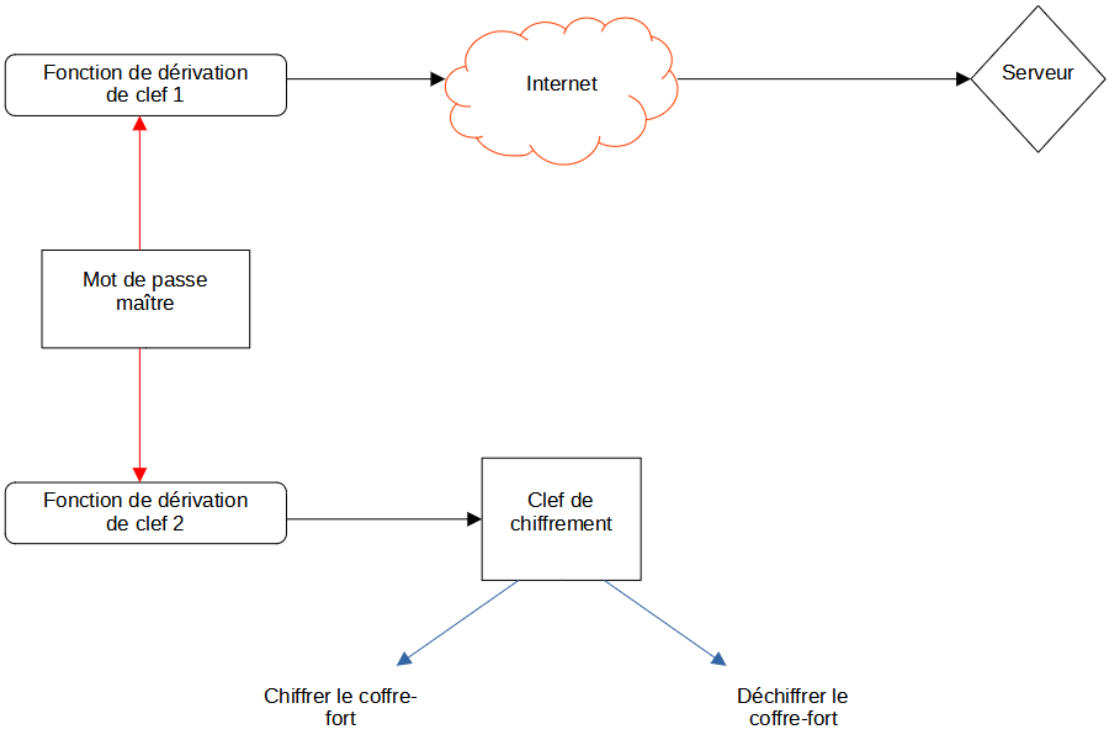


figure 1 : Résumé du fonctionnement général d'un gestionnaire de mots de passe

2) Les solutions de communication

Dans cette partie, nous allons détailler les différentes solutions de communication entre deux appareils qui permettent de partager les mots de passe.

Nous ferons une distinction entre les solutions utilisables localement, et celles qui permettent une synchronisation à distance :

- La **technologie Bluetooth**, permettant une communication à courte distance entre les appareils et permettant le transfert de données.

- L'utilisation de **QR Code** a aussi été prise en compte, permettant d'obtenir le lien pour la réception du fichier à partager. Le transfert ne peut cependant se faire que dans le sens "PC vers smartphone/tablette" et non "smartphone/tablette vers PC", du fait de la nécessité d'un scanner QR Code.

- La synchronisation **grâce à un support,** comme une clef usb ou une carte SD : l'appareil source met les données sur le support, on met le support sur l'appareil destination et on les récupère. On peut aussi le faire à l'aide d'un câble. Cette solution est très simple mais relativement contraignante pour l'utilisateur.

Un autre type de solution prend en compte le fait que les appareils à synchroniser soient distants. Pour ce cas de figure, nous avons dégagé différentes possibilités :

- L'utilisation du **protocole SFTP** (Secure File Transfer Protocol) peut se décliner en deux solutions. Soit nous utilisons l'appareil source comme un serveur pour envoyer au deuxième appareil, mais dans ce cas, il faudra faire une configuration NAT car il y a un trafic entrant, soit nous utilisons un serveur sftp dans le cloud depuis lequel les informations seraient rapatriées à la demande d'un utilisateur, mais cela nécessite encore un serveur.

- Le modèle de réseau **Peer to Peer** permettrait également de résoudre le problème de synchronisation : afin de récupérer le fichier cible, l'appareil va interagir avec les noeud du réseau qui possèdent une partie du fichier en interrogeant le serveur possédant la liste des noeuds et les fichiers qu'ils possèdent. Bien que ce modèle passe par l'utilisation d'un serveur pour guider les appareils vers le fichier cible, sa charge de travail reste réduite et peut sembler viable comme solution pour notre projet.

- La **Blockchain** est un modèle décentralisé qui se popularise de plus en plus dans différents domaines. Elle permet un partage de données décentralisé entre les nœuds qui contribuent à cette Blockchain. Les utilisateurs peuvent y stocker des données en les envoyant à un nœud qui va les vérifier puis les ajouter à la Blockchain. Tous les blocs de la Blockchain sont partagés avec l'ensemble des nœuds. Ce type de technologie présente malgré tout certains problèmes : la liste de mots de passe (dans notre cas), bien que chiffrée de façon rigoureuse, sera toujours présente sur la Blockchain, disponible par tous par définition même du principe de la Blockchain. Cela implique qu'elle sera à la merci d'individus pouvant tenter de décrypter nos informations (du fait que le fichier chiffré sera toujours disponible pour eux). La mise à jour de n'importe quel mot de passe de notre liste produira également à chaque fois un nouveau fichier chiffré de l'ensemble de nos mots de passe sur la Blockchain, toujours non supprimable. De plus, le principe de non-répudiation fait que le système garde une trace de qui a ajouté une information sur la Blockchain, laissant ainsi la gestion de la sécurité de notre anonymat dans leurs mains.

- L'utilisation du **protocole IPv6** est une voie possible pour réaliser la communication entre nos appareils distants. Le fait de passer par ce protocole permet une communication directe sans passer par le NAT, de par le nombre d'adresses disponibles. Cependant, si techniquement cela résout le problème, en réalité les NAT sont présents partout dans le monde et les adresses IPV6 sont parfois bloquées par les firewall, on ne peut donc pas utiliser cette solution dans tous les cas, en fonction de l'utilisateur.

- Les **solutions "stateless**" fonctionnent différemment : cette solution va combiner le mot de passe maître de l'utilisateur et les informations du site sur lequel

on veut se connecter (nom du site, URL,..) pour créer le mot de passe qui sera utilisé. Cela demande cependant de modifier l'ensemble de ses mots de passe utilisant ce type de solution et aura des limites dans le cas où l'on doit nécessairement utiliser le même mot de passe pour des sites différents, où le cas de sites nécessitant un mode d'authentification spécifique, notamment ceux demandant un format de mot de passe particulier (chiffres uniquement ou caractères spéciaux interdits). Dans ces cas-là, la solution stateless ne sera pas utilisable. De plus, si on doit changer le mot de passe maître, il est nécessaire de changer tous les mots de passe.

- Une solution basée sur **le protocole ICE(RFC5245)**. L'idée est d'utiliser ce protocole, couplé aux protocoles STUN(RFC5389) et TURN(RFC5766), afin que chaque appareil puisse découvrir si il se trouve derrière un NAT (dans le cas contraire on peut communiquer directement sans avoir de soucis liés au NAT), et si c'est le cas, leur fournir des adresses qu'ils pourront utiliser pour communiquer de pair à pair.

En premier lieu, chaque appareil interroge un serveur STUN/TURN, qui informera le client de s'il se trouve derrière un NAT, et si c'est le cas lui fournira deux adresses publiques utilisables, l'adresse du NAT le plus éloigné du client (i.e le NAT a priori visible sur internet), et une adresse publique du serveur. Le client dispose alors a priori de trois adresses utilisables : sa propre adresse IP, celle du NAT, et celle du serveur.

Ensuite, le couple d'adresse qui sera utilisé pour communiquer directement entre les appareils est négocié par le protocole ICE. En effet, parmi les six adresses en jeu, certaines ne pourront pas fonctionner dans tous les cas (les adresses IP des appareils seront des adresses privées s' ils sont situés derrière des NAT et ne pourront pas fonctionner par exemple). Une fois ceci fait, une connexion directe s'établit pour l'échange des mots de passe.

3) Conclusion sur les solutions les plus adaptées

Afin de pouvoir établir une comparaison entre les différentes solutions potentielles, nous avons défini certains critères tels que :

- la synchronisation locale, permettant de partager un fichier entre appareils proches
- la synchronisation distante
- la synchronisation facile, avec un ensemble d'actions simples à réaliser côté utilisateur
- l'utilisation d'une technologie accessible pour l'utilisateur, ne demandant pas ou peu de configuration manuelle pour l'utilisateur
- des échanges de données sécurisés, c'est-à-dire limitant la présence de nos mots de passe (même bien chiffrés) sur le réseau.

	Synchronisation locale	Synchronisation distante	Synchronisation facile	Technologie accessible pour l'utilisateur	Echange des données "sécurisé"
Réseau local					
Clé USB					
Bluetooth					
QrCode					
StateLess					
SFTP					
P2P					
BlockChain					
IPV6					
ICE					

Précisons que les solutions suivantes ne sont pas complètement à mettre de côté, elles présentent chacune des avantages non négligeables et pourraient être utiles dans certaines circonstances. Cependant, dans le cadre de ce projet, nous devons choisir les solutions les plus adaptées, c'est pourquoi il est nécessaire de faire une sélection appropriée.

Solutions écartées:

- Tout d'abord, les solutions qui ne permettent pas d'avoir un échange de données très sécurisé (sans accès aux données lors de la communication ou plus tard) sont écartées. Ainsi la BlockChain qui conserve les données éternellement (donc nos mots de passe chiffrés) ne peut pas être viable pour un projet de gestionnaire de mot de passe. La technologie Peer to Peer pourrait sembler viable pour notre projet, cependant, il se pose des questions autour de la configuration du NAT de la part du client Peer to Peer (client torrent par exemple) qui ne sont pas résolues.
- Ensuite les solutions difficiles d'utilisation comme SFTP qui nécessitent une configuration sur les appareils pour utiliser le protocole et peut nécessiter de passer par du NAT pour pouvoir fonctionner, rendent la mise en œuvre compliquée pour le grand public. On l'écarte donc.
  La solution IPV6, quant à elle, n'est pas accessible par un grand nombre de personnes pour l'instant donc on l'écarte aussi.
- La solution StateLess est intéressante mais la synchronisation reste difficile sans serveur. On peut l'écarter.
- La solution de l'utilisation du réseau local pour synchroniser des appareils est très intéressante pour synchroniser des appareils qui sont sur le même réseau.

Cependant, l'objectif ici sur ce projet est de trouver des alternatives à cette solution donc on ne s'y penchera pas.

- L'utilisation d'une clé USB ou autre support pour synchroniser des appareils est efficace mais contraignant pour l'utilisateur. De plus, l'intérêt dans ce projet de s'intéresser à ce type de synchronisation est assez bas donc on écarte cette solution.

Solutions retenues:

Nous allons retenir plusieurs solutions pour que l'utilisateur puisse utiliser la solution la plus adaptée à son cas de figure.

- QrCode:

  L'utilisateur va pouvoir synchroniser facilement ses mots de passe depuis un appareil (PC, smartphone, tablette) vers un appareil portable (smartphone ou tablette).

  Cette solution est simple pour synchroniser 2 appareils proches.
- Bluetooth:

  L'utilisateur va pouvoir synchroniser facilement ses mots de passe entre 2 appareils quels qu'ils soient (possédant quand même le bluetooth) à proximité.
- ICE:

  La solution a été retenue car elle permet une synchronisation à distance facile, de manière décentralisée car elle permet une communication pair à pair, elle remplit donc a priori tous les objectifs que nous nous sommes fixés.

IV) Définition des objectifs de l'étude expérimentale

Pour tester ces solutions envisagées, nous allons développer un programme qui va pouvoir synchroniser des mots de passe entre 2 appareils.

Le but est qu'un appareil puisse envoyer des mots de passe chiffrés via les solutions retenues (QrCode, Bluetooth, ICE) et qu'un second reçoive ces données.

On va alors réaliser une solution commune avec 3 modes (un pour chaque moyen de communication).

Cette solution aura plusieurs fonctionnalités :

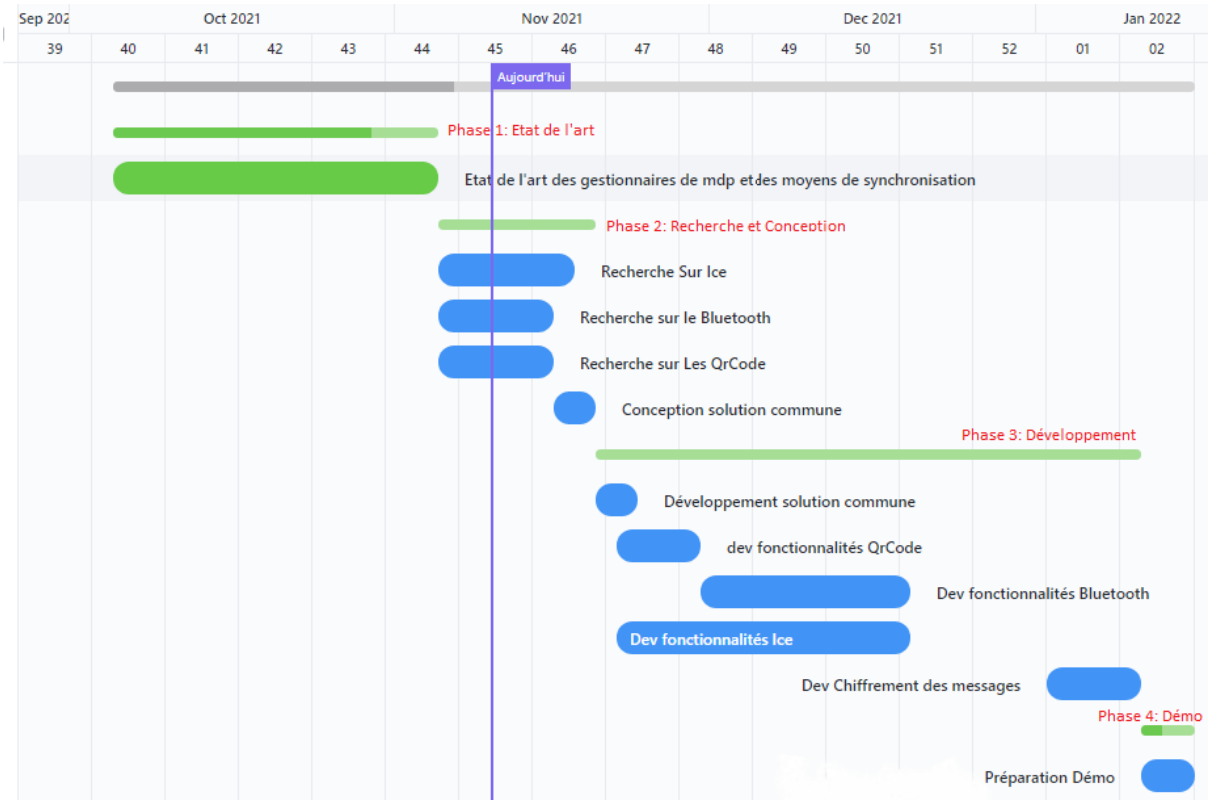
- Recherche d'un appareil pour communiquer.
- Envoyer des données.
- Recevoir des données.

Ces fonctionnalités seront disponibles pour les 3 modes que nous pourrons choisir au départ. Une solution commune aux 3 modes sera utile pour d'une part pour harmoniser notre code et d'autre part, si nous voulons rajouter des modes supplémentaires, il sera plus simple de les intégrer. De plus, le but est de faciliter l'utilisation par l'utilisateur qui n'aura qu'à choisir en fonction de son besoin, simplement en cliquant sur un bouton pour choisir son mode par exemple.

Pour finir, après que les différentes fonctionnalités soient prêtes, nous nous pencherons sur la mise en place d'un chiffrement de type gestionnaire de mot de passe, évoqué dans la partie III)1). La démonstration finale consistera à faire synchroniser des mots de passe entre 2 appareils et ce avec les 3 moyens possibles.

V) Planning du projet

Diagramme de Gantt du projet:



VI) Conclusion

L'objectif du projet est de fournir une solution de gestionnaire de mots de passe simple d'utilisation, permettant une synchronisation efficace, et de manière décentralisée. Pour cela, après avoir recensé les solutions potentielles, notre choix, après la prise en compte des différents avantages et inconvénients de chaque solution, s'est finalement porté sur 3 technologies : le **Bluetooth**, le **QR Code** et le **protocole ICE**.

La réalisation du projet se basera sur une **solution commune** permettant d'utiliser ces **3 moyens de communication** qui correspondront chacun à un **mode d'utilisation**. La solution sera dotée de fonctionnalités, qui pourront être étoffées en fonction de l'avancée du développement. L'aspect sécurité, mis en valeur lors de l'étude du fonctionnement d'un gestionnaire, sera pris en compte et développé une fois que les fonctionnalités définies pour chaque mode seront fonctionnelles.