Département des Technologie de l'information et de la communication (TIC)
Filière Télécommunications
Orientation Sécurité de l'information

Travail de Bachelor

Gestionnaires de mots de passe : quelle sécurité ?

Étudiante	**Noémie Plancherel**
Enseignant responsable	Prof. Sylvain Pasini
Année académique	2022-2023

Yverdon-les-Bains, le 26 octobre 2022

Département des Technologie de l'information et de la communication (TIC)
Filière Télécommunications
Orientation Sécurité de l'information
Étudiante : Noémie Plancherel
Enseignant responsable : Prof. Sylvain Pasini

Travail de Bachelor 2022-2023

Gestionnaires de mots de passe : quelle sécurité ?

Résumé publiable

Dans ce travail... Ceci est le résumé publiable...

Étudiant :	Date et lieu :	Signature :
Noémie Plancherel	.....	.....
Enseignant responsable :	Date et lieu :	Signature :
Prof. Sylvain Pasini	.....	.....

Préambule

Ce travail de Bachelor (ci-après TB) est réalisé en fin de cursus d'études, en vue de l'obtention du titre de Bachelor of Science HES-SO en Ingénierie.

En tant que travail académique, son contenu, sans préjuger de sa valeur, n'engage ni la responsabilité de l'auteur, ni celles du jury du travail de Bachelor et de l'Ecole.

Toute utilisation, même partielle, de ce TB doit être faite dans le respect du droit d'auteur.

HEIG-VD

Vincent Peiris
Chef de département TIC

Yverdon-les-Bains, le 26 octobre 2022

Authentification

La soussignée, Noémie Plancherel, atteste par la présente avoir réalisé ce travail et n'avoir utilisé aucune autre source que celles expressément mentionnées.

Yverdon-les-bains, le 26 octobre 2022

Noémie Plancherel

Cahier des charges

Résumé du problème

De nos jours, les gestionnaires de mots de passe sont des outils très fréquemment utilisés. En effet, une bonne pratique est d'utiliser un mot de passe par service. De cette manière, si un service est compromis et le mot de passe divulgué, cela n'impacte pas les autres services. Il est également très important de choisir un mot de passe fort qui ne contient pas d'éléments facilement prévisibles et qui pourrait être brute-forcé rapidement.

Les gestionnaires de mots de passe permettent principalement de faciliter le stockage des mots de passe qui demandent d'être de plus en plus longs et complexes, de manière à ne pas les réutiliser. Ils permettent également d'ajouter une couche sécuritaire aux mots de passe en les stockant de manière sécurisée et en offrant la possibilité de générer des mots de passe forts.

Ces applications offrent plusieurs fonctionnalités sous la forme de différents types ; elles permettent, entre autres, l'utilisation du cloud afin de stocker les mots de passe sur les serveurs du fournisseur pour faciliter la synchronisation des données entre plusieurs devices (mobile, montre, navigateur, etc.). Certains gestionnaires de mots de passe sont également fréquemment utilisés au sein d'entreprises pour permettre le partage de données. Les entreprises vont généralement utiliser une solution self-hosted où elles auront leur propre infrastructure et stockage. Il existe des extensions de navigateur qui proposent le remplissage automatique de mots de passe dans les formulaires de connexion. Enfin, il y a également des applications en local qui vont limiter leur utilisation à un seul appareil.

Étant donné que les utilisateurs se reposent grandement sur les gestionnaires de mots de passe, il est important de s'assurer que ces logiciels satisfassent un certain nombre de principes de sécurité ainsi qu'une implémentation robuste afin d'éviter tout vol ou perte de données.

Objectifs

Ce travail de Bachelor vise à comprendre les menaces d'un gestionnaire de mots de passe, premièrement de manière générique, puis sur des produits spécifiques, sélectionnés à la suite d'une étude complète, en analysant la sécurité sous différents angles (stockage, mémoire, réseau, cryptographie, etc.).

Le travail est réalisé en deux parties distinctes ; une première partie qui est une étude approfondie et complète sur les gestionnaires de mots de passe. Elle permet d'analyser les menaces des différents type de gestionnaires de mots de passe et de présenter les exigences sécuritaires qu'il serait nécessaire de garantir. Elle va également se concentrer sur une étude de marché avec une comparaison de plusieurs gestionnaires de mots de passe existants sous différents aspects.

La deuxième partie du travail se concentrera tout d'abord sur la sélection de quelques candidats (environ 4) en fonction de critères établis au préalable. Ensuite, le but est d'évaluer la sécurité de manière complète de chaque gestionnaire de mot de passe sélectionné ; chaque élément choisi est analysé et évalué en fonction de différents critères comme les choix cryptographiques utilisés, le stockage, ou encore l'architecture de l'application.

Livrables

Les délivrables seront les suivants :

1. Une documentation contenant :
   - Présentation des différents types de gestionnaires de mots de passe
   - Étude de marché
   - Une analyse de menaces de différents types de gestionnaires de mots de passe
   - Spécification des exigences sécuritaires à garantir

   (a) Analyse sécuritaire des quelques candidats représentatifs (environ 4) :
   - Sélection de candidats pour la suite du travail

   chaque analyse se décomposera ainsi :
   - Sélection de critères d'analyse
   - Analyse complète de chaque aspect
   - Rapport des faiblesses trouvées au fabricant

   (b) Synthèse des résultats
2. Comparaison entre chaque candidat

Déroulement

En se référant aux dates validées par M.Donini, le travail de Bachelor débute le 20 septembre 2022 et se termine au plus tard le 10 février 2023. Il y a 3 dates clés incluant des rendus :

- **14 octobre 2022** - rendu du rapport intermédiaire
- **14 décembre 2022** - rendu du rapport final
- **23 janvier au 10 février 2023** - soutenance du travail de bachelor

Etant donné, que la soutenance du travail implique l'intervention d'un expert, la date doit être définie entre tous les intervenants.

Le volume du travail de bachelor est de 15 crédit ECTS, soit 450 heures. Le rapport intermédiaire représente 150 heures de travail.

Au niveau de la répartition de la charge de travail, cela représente environ 45h/semaine jusqu'au rendu, soit le 14 décembre, car le travail se fait à 100%.

Planning

Le travail de bachelor sera séparé en plusieurs tâches et sous-tâches différentes qui permettront de répartir plus facilement le travail sur des périodes de plusieurs semaines. Ci-dessous, le planning détaillé avec toutes les tâches :

1. Préparation
    - Rédaction du cahier des charges
    - Planification
    - Recherches initiales et introduction
2. Étude de marché
    - Recherche et explication des différents types de gestionnaires de mots de passe
    - Comparaison des fonctionnalités, du prix et des plateformes disponibles
    - Analyse du marché actuel et de la demande
    - Récapitulatif
3. Étude sécuritaire
    - Présentation de la sécurité implémentée dans les gestionnaires de mots de passe
    - Identification et analyse des menaces potentielles
    - Rédaction des exigences sécuritaires
4. Sélection
    - Mise en place des critères de sélection des candidats
    - Sélection des candidats
5. Analyse sécuritaire (pour chaque candidat)
    - Identification et rédaction des critères d'analyse
    - Analyse sécuritaire de chaque aspect
6. Synthèse (pour chaque candidat)
    - Synthèse des résultats
    - Rapport des faiblesses au fabricant

7. Synthèse générale
    - Comparaison de tous les résultats
    - Conclusion du travail
8. Documentation
    - Rédaction du rapport
    - Lecture / visualisation de documents
    - Tenue d'un journal de travail

Un diagramme de Gantt a également été effectué afin de pouvoir visualiser le planning et ajouter des périodes de temps :

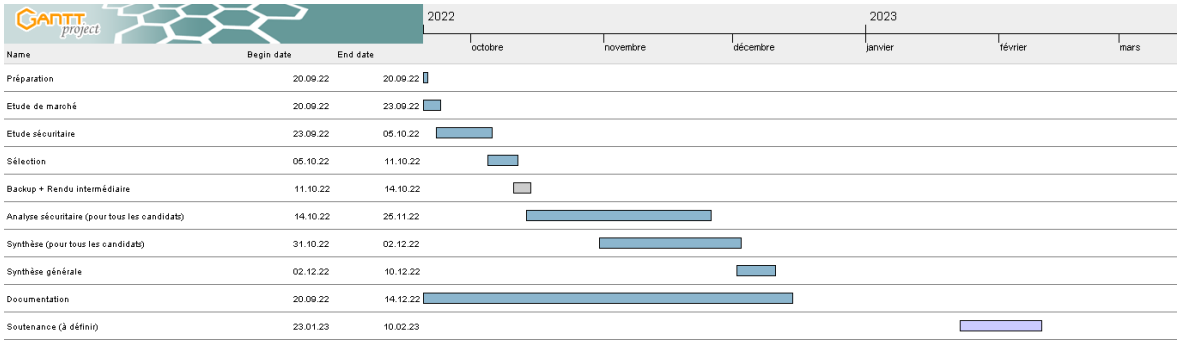

FIGURE 1 – Planning du travail de Bachelor

Table des matières

Chapitre 1

Introduction

Pour un utilisateur lambda, il peut être difficile de se souvenir de tous ses mots de passe tout en s'assurant d'en utiliser un différent pour chaque service afin d'éviter tout vol de données. Typiquement dans ces situations, nous allons naturellement utiliser des mots de passe simples, qui sont facilement mémorisables. Comme, par exemple, utiliser son prénom et sa date de naissance, "123456" ou encore "qwerty". De plus, il est plus simple d'utiliser le même mot de passe pour chacun de ses comptes, afin d'en mémoriser uniquement un seul.

Cependant, même si l'unique mot de passe qu'on utilise est fort et aléatoire, il n'est pas garanti à 100% qu'on soit la cible d'aucun attaquant et si une attaque est réalisée, toutes nos données personnelles sont exposées.

Ainsi, dans ce genre de cas, les gestionnaires de mots de passe interviennent et peuvent faciliter le quotidien de la plupart des utilisateurs.

1.1 Fonctionnement général

Les gestionnaires de mots de passe sont des applications multi-plateformes qui vont permettre de stocker des informations sensibles telles que des mots de passe, numéros de carte de crédit ou encore des fichiers confidentiels. On peut les comparer à des coffres forts.

Ces derniers proposent un *master password* ou une *master key* qui va permettre d'accéder à l'ensemble des données secrètes. En conséquence, la sécurité repose sur un seul mot de passe principal, ce qui est très bénéfique pour les utilisateurs car ils n'ont qu'un mot de passe à retenir. Une fois l'accès à l'application, l'utilisateur a la possibilité de stocker des données, générer des mots de passe ainsi que se connecter à des services en ligne (remplissage de formulaire d'identification automatique).

Les gestionnaires de mots de passe sont disponibles en plusieurs types différents en fonction

du besoin de l'utilisateur et des fonctionnalités proposées.

1.2 Types

1.2.1 Cloud

Les gestionnaires de mots de passe dans le cloud sont proposés pour un usage personnel ainsi qu'un usage professionnel. Les mots de passe entrés dans le coffre fort vont directement être stockés sur les serveurs du constructeur et ils seront également chiffrés sur ces derniers. Aucun stockage n'est effectué en local.

Le cloud va permettre aux utilisateurs d'avoir accès à leurs données sur n'importe quel device (ordinateur, mobile, montre) et à tout moment. De plus, toutes les données vont être synchronisées sur tous les devices connectés.

À propos de la sécurité, elle repose entièrement sur le provider de l'application car toutes les informations sont stockées sur leurs propres serveurs.

1.2.2 Local

Les applications en local s'installent sur le desktop ou sur le mobile de l'utilisateur. Ces gestionnaires de mots de passe fonctionnent indépendamment et sont offline. Ces produits peuvent donc être utilisés sur une seule machine, par conséquence la synchronisation n'est pas proposée pour ce type de password manager.

Toutes les données sensibles sont directement stockées et chiffrées sur le device. La sécurité est plutôt bonne comparé à la solution cloud car c'est du hors-ligne, cependant si on récupère / vole le device, la sécurité devient plus faible car il y aurait la possibilité d'avoir accès aux informations sensibles du gestionnaire de mots de passe, via notamment une gestion de mémoire mal gérée.

Il y a également une solution *on-premise* (ou *self-host*) qui permet d'utiliser sa propre infrastructure locale pour héberger toutes les données du gestionnaire de mots de passe. Les fonctionnalités offertes sont les mêmes que pour les solutions cloud mais le prix est en général plus cher et l'application plus orientée professionnelle.

1.2.3 Navigateur

Le dernier type de gestionnaire de mots de passe sont ceux qui sont basés sur le navigateur (*browser-based*). Les navigateurs les plus populaires, tels que Firefox, Safari ou Chrome offrent ce gestionnaire de mots de passe qui est directement inclu dans ces derniers. Ils vont faciliter la gestion et la sauvegarde de mots de passe de comptes de sites web. Il y a également

la possibilité de synchroniser toutes les données stockées entre tous les devices qui supportent le navigateur en question (Chrome, Firefox, Safari, etc.).

Pour certains navigateurs, les informations sont stockées et chiffrées en local sur le device de l'utilisateur. Si la synchronisation est activée, les données seront également stockées dans le cloud sur les serveurs du constructeur. Un problème de sécurité importante, et la disponibilité des mots de passe sur le navigateur, si aucun master password n'est configuré et qu'on a accès à la machine, les mots de passe sont accessibles en clair sur le navigateur.

Chapitre 2

Étude de marché

Ce chapitre vise à étudier les différentes fonctionnalités offertes par les gestionnaires de mots de passe en les comparant entre plusieurs produits sélectionnés et en établissant un tableau afin d'avoir une meilleure vue d'ensemble.

Nous allons également analyser les différents prix des applications ainsi que présenter où en est le marché actuel afin d'étudier la popularité de ces dernières.

Pour l'étude de marché, les gestionnaires de mots de passe sélectionnés seront : *LastPass*[1], *Dashlane*[2], *1Password*[3], *KeePass*[4], *Bitwarden*[5], *NordPass*[6], *Padloc*[7], *Keeper*[8], *Firefox*[9].

Ils ont été sélectionnés en se basant sur leur popularité sur le marché ainsi qu'à la suite de lecture d'articles concernant les meilleurs gestionnaires de mots de passe [5][8][18][15]. Les applications open-source ont été avantagées lors de leurs sélections.

2.1 Fonctionnalités

Ci-après, une liste des fonctionnalités disponibles dans les gestionnaires de mots de passe. Cette énumération se base sur toutes les fonctionnalités citées sur les websites des différents *password managers*.

1. https://www.lastpass.com/
2. https://www.dashlane.com/
3. https://1password.com/
4. https://keepass.info/
5. https://bitwarden.com/
6. https://nordpass.com/
7. https://padloc.app/
8. https://www.keepersecurity.com/
9. https://www.mozilla.org/fr/firefox/features/password-manager/

1. Stockage d'informations personnelles (cartes de crédit, passeport, contrats, etc.)
2. Remplissage automatique des formulaires en ligne (auto-complétion)
3. Partage de données entre plusieurs utilisateurs (par exemple, partage d'informations d'identifications entre une famille)
4. Générateur de mots de passe forts
5. Surveillance de la fuite de données ou données compromises
6. Alerte en cas de données compromises
7. Synchronisation de données entre devices (cloud)
8. Authentification à double facteurs
9. Self-hosting
10. Support prioritaire
11. Connexion à l'aide de facteurs biométriques (*fingerprint* ou *facial recognition*) ou d'un pin
12. *Backup & Restore*, possibilité de récupérer une ancienne sauvegarde

Ci-dessous un tableau récapitulatif qui indique quels gestionnaires de mots de passe offre quelles fonctionnalités.

Application	1	2	3	4	5	6	7	8	9	10	11	12
LastPass	×	×	×!	×*	×*	×*	×	×		×*	×*	
Dashlane	×*	×	×!	×	×*	×	×*	×!				×
1Password [10]	×*	×*	×*	×*		×*	×*	×*		×*		×
KeePass [11]	×	×		×	×			×	×		×	×
Bitwarden		×	×!	×	×*	×*	×	×!	×*	×*	×!	×
NordPass	×	×	×*	×	×*		×	×	×*	×*	×	
Padloc	×	×	×*	×	×*		×	×*	×	×*	×	
Keeper [12]	×*	×* [13]	×*	×*	×*	×*	×*	×*	×*	×*	×*	×*
Firefox		×		×	×	×	×	×				

TABLE 2.1 – Fonctionnalités proposées par les candidats

× : L'application propose cette fonctionnalité
***** : Fonctionnalité proposée mais avec un version premium (payante)
! : Limitations avec une version gratuite

10. L'application est totalement payante et différents abonnements sont proposés
11. En général, nécessite l'installation de plugins supplémentaires afin de profiter de toutes les fonctionnalités
12. Se référer à la note de bas de page 10
13. Extension *KeeperFill*

Sur tous les candidats sélectionnés, nous remarquons que la plupart offre la majorité des fonctionnalités énumérées plus haut. Nous constatons que l'offre des constructeurs de gestionnaires de mots de passe est assez variée et répond à la demande des particuliers et des entreprises.

Par rapport aux restaurations de sauvegarde, les gestionnaires cloud-based vont automatiquement créer des backups toutes les nuits, donc la restauration se fait directement dans le gestionnaire. Pour bitwarden, lorsque le gestionnaire est hébergé on-premise, il est nécessaire de créer ses propres procédures de sauvegardes. Etant donné que KeePass est uniquement en local, les sauvegardes doivent être faites manuellement et peuvent être importées sur l'application. Toutes ces solutions nécessitent le master password. Si ce dernier est oublié, il existe plusieurs solutions différentes en fonction des constructeurs (fonctionnalité pas disponible sur KeePass).

La "sauvegarde" qui ne nécessite pas d'avoir son master password est l'exportation des données en fichier CSV, mais à moins de chiffrer le fichier, les données sont en claires ce qui n'est évidemment pas sécurisé et pas très recommandé.

2.2 Plateformes

Cette partie va permettre de visualiser sur quelles plateformes les gestionnaires de mots de passe sélectionnés sont supportés.

Application	Windows	MacOS	Linux	Android	iOS	Navigateur
LastPass	×	×	×	×	×	×
Dashlane						×
1Password	×	×	×	×	×	
KeePass	×	×*	×*	×*	×*	×*
Bitwarden	×	×	×	×	×	×
NordPass	×	×	×	×	×	
Padloc	×	×	×	×	×	×
Keeper	×	×	×	×	×	×
Firefox						×

TABLE 2.2 – Plateformes supportées par les différentes applications

× : L'application est supportée sur ces plateformes
* : Des applications (ou des paquets) compatibles avec KeePass Password Safe non-officielles mais contribuées existent

Même si un gestionnaire supporte toutes les plateformes indiquées, il est nécessaire d'aller vérifier les conditions d'utilisation du système, c'est-à-dire les versions des plateformes afin de s'assurer que l'application fonctionnera quand même.

Cependant, nous constatons que la majorité des applications sont disponibles sur les plateformes les plus courantes, et même si elles ne le sont pas, il y a souvent une solution non-officielle (notamment pour KeePass) ou via le navigateur qui existe.

2.3 Prix

Nous allons passer brièvement en revue les prix proposés par les gestionnaires de mots de passe. Chaque application propose leurs propres gammes de prix avec également des abonnements possibles pour les particuliers, familles ou entreprises.

2.3.1 Particuliers

Pour la plupart des applications, nous pouvons retrouver 3 gammes de prix ; Gratuit, Premium, Famille. L'offre familiale va être plus chère car les gestionnaires de mots de passe sont conçus pour pouvoir avoir plusieurs gestionnaires chiffrés individuels.

Les tarifs ci-dessous sont exprimés en mensualités et en USD.

Application	Gratuit	Premium	Famille
LastPass	$0	$3	$4
Dashlane	$0	$3.99	$5.99
1Password	non	$2.99	$4.99
KeePass [14]	$0	non	non
Bitwarden	$0	<$1	$3.33
NordPass	$0	$1.84	$4.99
Padloc	$0	$3.49	$5.95
Keeper	non	$2.92	$6.25
Firefox	non	non	non

TABLE 2.3 – Tarifs pour particuliers

14. Gratuit et open-source

2.3.2 Entreprises

Les entreprises ont quant à elle des prix différents dû à leurs besoins spécifiques où ils pourraient avoir besoin d'un devis personnel afin de choisir l'abonnement qui convient au mieux à leur infrastructure. Il existe plusieurs catégories qui sont en fonction du nombre d'employés et également par rapport aux fonctionnalités souhaitées.

Chaque prix est indiqué en mensualités, en USD et par employé.

Application	Team	Business
LastPass	$4	$6
Dashlane	$5	$8
1Password	non	$7.99
KeePass [15]	non	non
Bitwarden	$3	$5
NordPass	non	$3.50*
Padloc	$3.49	$6.99*
Keeper	non	$3.75*
Firefox	non	non

TABLE 2.4 – Tarifs pour les entreprises

* : Il y a la possibilité d'établir un devis en fonction des besoins spécifiques de l'entreprise

2.4 Marché actuel

Afin d'effectuer une étude un peu plus approfondie et afin d'établir un constat de la demande actuelle sur le marché et de leur popularité, nous allons analyser les différentes statistiques des gestionnaires de mots de passe.

Malgré les multiples fonctionnalités proposées par les *password managers*, les particuliers restent plutôt réticents à l'idée d'en utiliser un régulièrement ; d'après un sondage lancé par PasswordManager[22] aux Etats-Unis avec des personnes âgées de 18-55+, seulement 22.5% utilisent des gestionnaires de mots de passe. Une autre étude de Security.org[24] de novembre 2021, également lancée aux Etats-Unis, ressort les mêmes statistiques ; 20% des utilisateurs utilisent ces derniers. Les autres solutions communes pour stocker ses identifiants sont la mémorisation, le papier, la réutilisation, etc. Nous pouvons sans aucun doute déclarer que ces méthodes ne sont pas très sécurisées.

15. Voir 14

Néanmoins, nous pouvons expliquer cette réticence à l'aide des études citées ci-dessus qui déclarent qu'au niveau des utilisateurs qui n'utilisent pas de gestionnaires de mots de passe, 70% ne font pas confiance à la sécurité qu'elles fournissent, ils pensent que leur application pourrait être hackée. Certains, ne font également pas confiance aux constructeurs de ces dernières en pensant qu'ils volent leurs données.

En contradiction à ces avis populaires, en se basant sur un sondage de 2022 de bitwarden[4], globalement, 35% des utilisateurs sont plus inquiets des cyberattaques par rapport à l'année 2020. Nous pouvons justifier ces inquiétudes avec le fait que le nombre de cyberattaques effectuées en 2021 a augmenté (en partie dû au COVID-19 et au *home office*). Le DBIR de 2022 (Data Breach Investigations Report)[25] indique qu'il y a eu une augmentation de 13% des vols de données (dont 85% font partie de vulnérabilités humaines).

Avec toutes ces statistiques, nous pouvons constater que malgré une utilisation encore trop basse des gestionnaires de mots de passe, les particuliers s'y intéressent progressivement dû aux attaques et vols de données en progression constante. Cependant, il y a un manque de confiance général sur ces derniers, surtout envers les constructeurs. C'est pourquoi, la sécurité parfaite au sein des gestionnaires est un sujet très important si l'on souhaite augmenter la protection des données et éviter des vulnérabilités humaines (notamment l'utilisation de mots de passe trop de simple, comme "123456"). La sécurité "presque" parfaite des applications pourraient également baisser les vols de données par des personnes malveillantes.

2.5 Récapitualtif de l'étude

Nous allons résumer toutes les informations que nous avons recueillies dans ce chapitre-ci ; au final, nous constatons que les gestionnaires de mots de passe qui sont actuellement sur le marché (ici les plus populaires), sont assez complets au niveau des fonctionnalités proposées et ils sont adaptées pour tout type d'utilisation (personnelle, familiale ou professionnelle). Pour les gestionnaires payants, leurs prix sont assez abordables pour l'offre qu'ils proposent. Toutefois, même les gestionnaires en version gratuite, convient tout à fait à une utilisation quotidienne.

Au niveau des applications comparées, toutes ont leurs points positifs et leurs points négatifs (l'aspect sécuritaire sera discuté dans le chapitre *étude sécuritaire*).

LastPass propose une version gratuite avec les fonctionnalités classiques que l'on attend d'un gestionnaire de mot de passe. La limite est que l'application n'est accessible que depuis un seul type d'appareil, ils font la différence entre ordinateur (fixe et portable) et appareil mobile (téléphone, montre, tablette). La version payante offre le MFA ainsi qu'un dashboard (avec les alertes de sécurité et la surveillance sur les données compromises), ce qui est une fonctionnalité intéressante.

La version gratuite de DashLane propose un stockage jusqu'à 50 mots de passe, ce qui est au

final assez limtité mais il offre le 2FA ainsi que le partage sécurisé (jusqu'à 5 comptes). La version premium, permet l'utilisation d'un VPN ainsi qu'une synchronisation sur plusieurs appareils.

1Password est totalement payant mais est l'un des gestionnaires de mots de passe le plus populaire sur le marché.

KeePass est une application gratuite et open-source. Il propose une grande sélection de plugins assez utiles et variés, ce qui permet une grande offre, en plus d'être complètement gratuite.

bitwarden propose une version gratuite étonnement très complète avec un stockage illimité de mots de passe et un nombre illimité d'appareils. La version premium offre un 2FA avancé (notamment la connexion à l'aide d'une Yubikey) ainsi que des rapports de sécurité.

NordPass propose également une version gratuite complète qui permet le MFA ou encore la synchronisation entre plusieurs appareils, ce qui est très utile. Le premium propose l'aspect sécuritaire en plus. Cependant, c'est la solution gratuite la meilleure de tous les candidats sélectionnés.

Padloc n'est pas un gestionnaire très populaire mais il l'avantage d'être open-source et d'être disponible sur Github [16]. La version gratuite n'offre pas beaucoup de fonctionnalités mais il y a la possibilité de stocker un nombre illimité de secrets et d'y connecter un nombre illimité d'appareils. De plus, il est multi-plateformes.

Keeper est complètement payant mais a une offre très complète et est particulièrement bien adapté pour les entreprises.

Finalement, le gestionnaire de mots de passe proposé par Firefox est directement inclus avec le navigateur, ainsi ses fonctionnalités proposées sont assez basiques et pas très poussées, mais il propose les fonctionnalités attendues d'un gestionnaire, c'est-à-dire enregistrement, génération et synchronisation de mots de passe.

16. https://github.com/padloc/padloc

Chapitre 3

Étude sécuritaire

Ce chapitre est dédié à toute l'analyse sécuritaire des gestionnaires de mots de passes en général. Nous allons dans un premier temps décrire comment ces applications sont sécurisées en fonction de leur type, puis présenter les différents algorithmes utilisés dans les candidats sélectionnés

3.1 Implémentation de la sécurité dans les gestionnaires de mots de passe

Dans cette section, afin de se baser sur des gestionnaires de mots de passe déjà existants et de pouvoir comparer les différentes sécurités implémentées, nous allons reprendre les 9 candidats sélectionnés dans la partie *étude de marché*, c'est-à-dire ; *LastPass*, *Dashlane*, *1Password*, *KeePass*, *Bitwarden*, *NordPass*, *Padloc*, *Keeper* et *Firefox*. Toutes les informations citées sont basées sur les *security whitepapers* des constructeurs[16][6][1][13][3][21][14][19].

Les gestionnaires de mots de passe sélectionnés fonctionnent tous de la même manière, au final cette méthode est plutôt classique dans les architectures des applications. Un *master password* (qui est seulement connu par l'utilisateur) est généré ou entré par l'utilisateur et va permettre le déverrouillage de l'application et le chiffrement / déchiffrement en local de toutes les données stockées.

À part pour les gestionnaires en local qui gèrent la sécurité différemment, ils mettent en avant le *Zero-knowledge encryption*. C'est une méthode qui va permettre un chiffrement *end-to-end* et qui va sécuriser au mieux les données personnelles et sensibles des utilisateurs, des serveurs du constructeur. En sachant que toutes les données sont stockées dans le cloud du provider, afin d'éviter que n'importe qui puisse y avoir accès, toutes les données sont chiffrées avant d'être envoyées au serveur. Ainsi, les données sont uniquement déchiffrées sur le device de l'utilisateur et la clé de chiffrement reste également en local.

Nous allons décrire dans les sous-sections suivantes comment la sécurité est implémentée dans les gestionnaires de mots de passe en fonction de leur type afin d'aller un peu plus en détail.

3.1.1 Les gestionnaires browser-based

Les gestionnaires de mots de passe *browser-based* proposent des fonctionnalités classiques et ne sont pas très poussés. Ce sont des applications légères qui sont pensées pour faciliter au mieux les utilisateurs.[26]

Par défaut, ces applications n'activent pas de *master password* et certaines n'en proposent même pas pour ajouter un chiffrement supplémentaire. Toutefois, ils proposent la fonctionnalité de 2FA.

Firefox fonctionne localement ou avec la synchronisation qui demande l'utilisation du cloud. Si la synchronisation n'est pas activée, les mots de passes stockés sont directement chiffrés sur l'appareil et sont ajoutés à un fichier *logins.json* qui se trouve dans le répertoire de l'utilisateur.

Il propose également la fonctionnalité d'ajouter un master password afin d'avoir une couche sécuritaire supplémentaire (par défaut, cette fonction est désactivée), ainsi, sans un mot de passe principal et sans l'activation du 2FA, les mots de passe sont accessibles en clair sur le navigateur dès le moment où on a accès à l'appareil et que l'utilisateur est connecté à son compte (ce qui est fortement probable, car le compte reste connecté, même après une fin de session). Toutefois, lors de l'ajout d'un master password, ce dernier est uniquement défini en local et n'est pas synchronisé entre profils ou appareils, mais il chiffrera toutes les données en local.

Au niveau du fonctionnement[2] (sans master password), les identifiants sont chiffrés à l'aide de 3DES-CBC et sont directement stockés dans un fichier JSON *logins.json* encodés en ASN.1 puis en Base64. La clé de chiffrement est stockée dans une base de donnée *key4.db*. Il existe actuellement des outils pour déchiffrer les mots de passe du gestionnaire.

Si l'option de synchronisation (*Firefox Sync*) est activée[23], les données stockées sur les serveurs Mozilla seront chiffrés. Nous pouvons se baser sur le schéma suivant qui explique le processus de synchronisation des données avec les serveurs Mozilla :

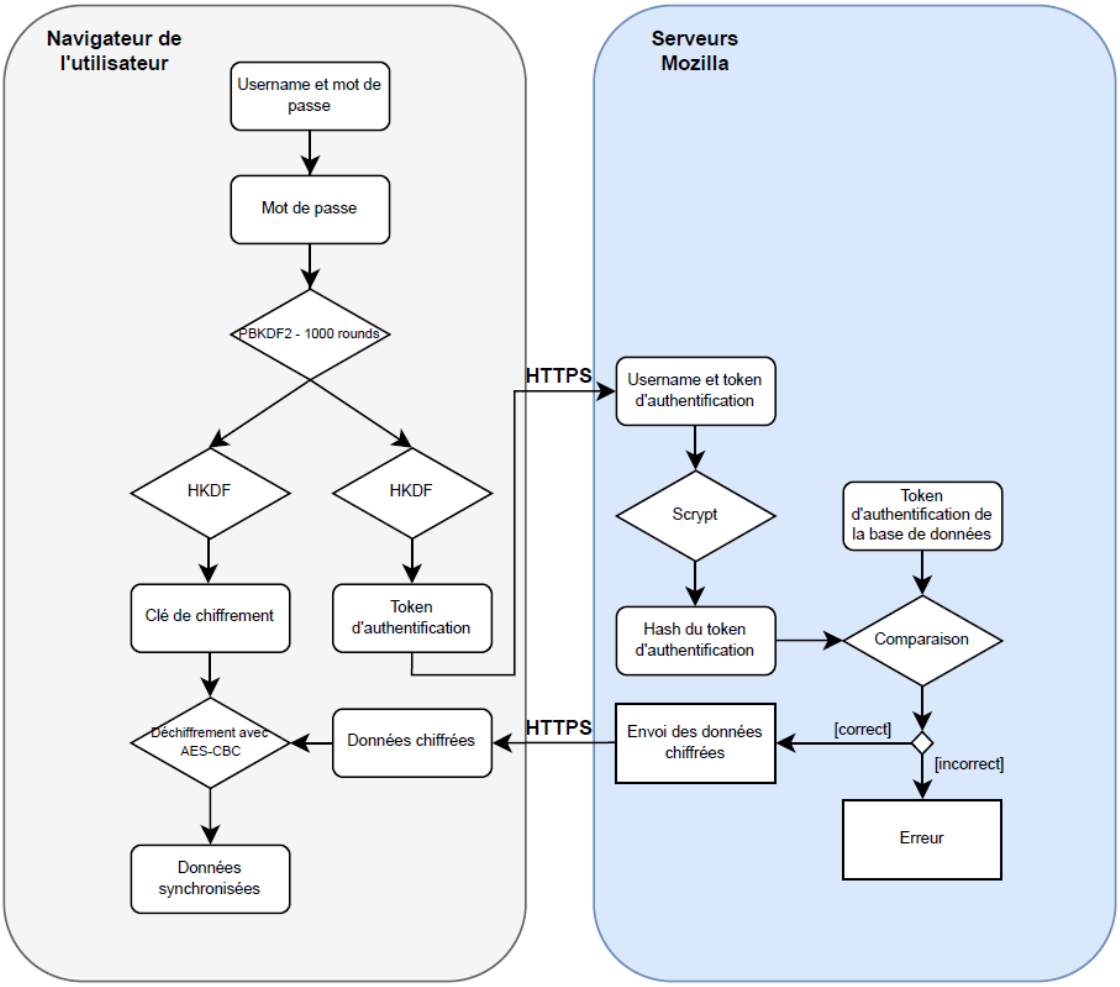

FIGURE 3.1 – Schéma de synchronisation de données sur Firefox

Les données chiffrées sont protégées avec HMAC-SHA256 afin de pouvoir authentifier les données avant de les déchiffrer. La méthode utilisée est Encrypt-then-MAC. Cela permet de protéger l'intégrité des données en clair et chiffrées.

Comme cité précédemment, lors d'un ajout de master password afin de protéger les données localement, Firefox va tout d'abord chiffrer les données localement (le stockage se fait comme expliqué plus haut) et va les rechiffrer lors de la synchronisation avec le processus expliqué à l'aide du schéma.

Au final, ce design cryptographique est plutôt classique au sein des gestionnaires de mots de passe qui utilisent les serveurs du constructeur pour stocker les données.

Les autres gestionnaires de mots de passe browser-based fonctionnent en général de la même manière avec les données synchronisées. Chrome stocke également les identifiants dans un fichier local protégé avec DPAPI (Microsoft's Data Protection API).

3.1.2 Les gestionnaires en local

Pour les gestionnaires qui fonctionnent uniquement en local, nous allons nous baser sur l'implémentation sécuritaire de KeePass, qui est open-source, ce qui peut faciliter à comprendre tout le concept. Dans cette section, nous n'allons pas détailler toute l'architecture mais rester plutôt en surface. D'autres gestionnaires de mots de passe fonctionnent également en *offline* avec un stockage local, cependant lors de la première connexion, il y a quand même des informations envoyées aux serveurs du constructeur (comme 1Password par exemple), ce qui n'est pas le cas pour l'application de base KeePass.

Etant donné que KeePass ne fonctionne qu'en local, il est important de bien gérer la mémoire et le stockage de données sensibles.

Toutes les données se trouvent dans un fichier de base de données spécial *.kdbx*. Ce fichier contient toutes les données du gestionnaire (mots de passe, usernames, etc.) et est chiffré (également compressé en GZIP si on le souhaite). Dans la version de KeePass 2.x, les chiffrements supportés sont AES256-CBC et ChaCha20.

La base de données est stockée où l'utilisateur le souhaite (avec une possibilité de la stocker dans un cloud), c'est pourquoi la sécurité repose sur la complexité du mot de passe. Elle est structurée avec un header et un contenu[17][10].Dans le header, sont stockées différentes informations ; un UUID indiquant le cipher, une indication si le fichier est compressé, différentes seed (voir 3.2), l'IV pour le chiffrement et des bytes pour l'authentification (générés aléatoirement lors de la sauvegarde de la base de données). Le corps du fichier contient des blocs hachés avec HMAC-SHA256 et des blocs de données qui ont un format XML lorsqu'ils sont déchiffrés.

Tous les deux sont authentifiés avec le schéma Encrypt-then-Mac avec HMAC-SHA256.

Au niveau de l'utilisation de la mémoire du processus, lors de l'ouverture de la base de données chiffrée, elle est chargée en mémoire. La protection de la mémoire s'applique aux données sensibles telles que la Master key et des mots de passe. Dès que la base de donnée est sauvegardée, les données chiffrées sont envoyées sur le disque. Dès le moment où l'application est verrouillée ou le processus arrêté, le disque est nettoyé afin qu'il ne contienne aucune information sensible.

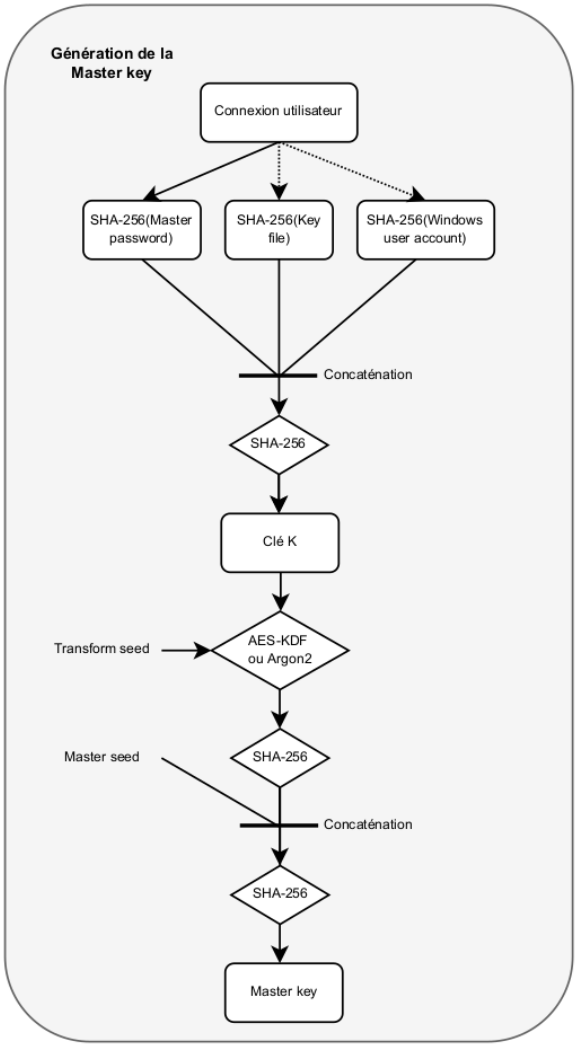

FIGURE 3.2 – Génération de la Master key sur KeePass

La Master key sur KeePass prend plusieurs arguments différents et est générée d'une manière assez complexe. L'utilisateur doit obligatoirement fournir un master password, et peut également utiliser un key file et / ou le compte utilisateur Windows.

Chaque composant est haché avec SHA-256 et sont concaténés et hachés ensemble. En output, on a une clé composite K qui va être dérivée à l'aide de AES-KDF ou Argon2. Les paramètres de ces deux algorithmes peuvent être configurés dans les paramètres de la base de données.

Chaque output est haché avec SHA-256 et au final, on obtient la Master key qui permettra de déchiffrer la base de données.

Les deux différentes seeds utilisées dans la génération de la clé sont stockées dans le header de la base de données *.kdbx*.

Cette architecture permet de se protéger contre les attaques par dictionnaires et le brute-force de la Master key.

3.1.3 Les gestionnaires cloud-based

Les gestionnaires de mots de passes cloud-based ont tous une architecture similaire dû au fait que toutes les données sont stockées sur les serveurs du constructeur. Il y a des différences avec l'authentification de l'utilisateur et des algorithmes cryptographiques choisis, mais le design sécuritaire a la même base au niveau de la connexion de l'utilisateur et du chiffrement du gestionnaire.

Ainsi, afin d'expliquer un peu plus en détail le fonctionnement d'un gestionnaire cloud-based, nous allons prendre l'exemple de LastPass. Nous allons nous basé sur le schéma ci-dessous afin d'appuyer nos propos.

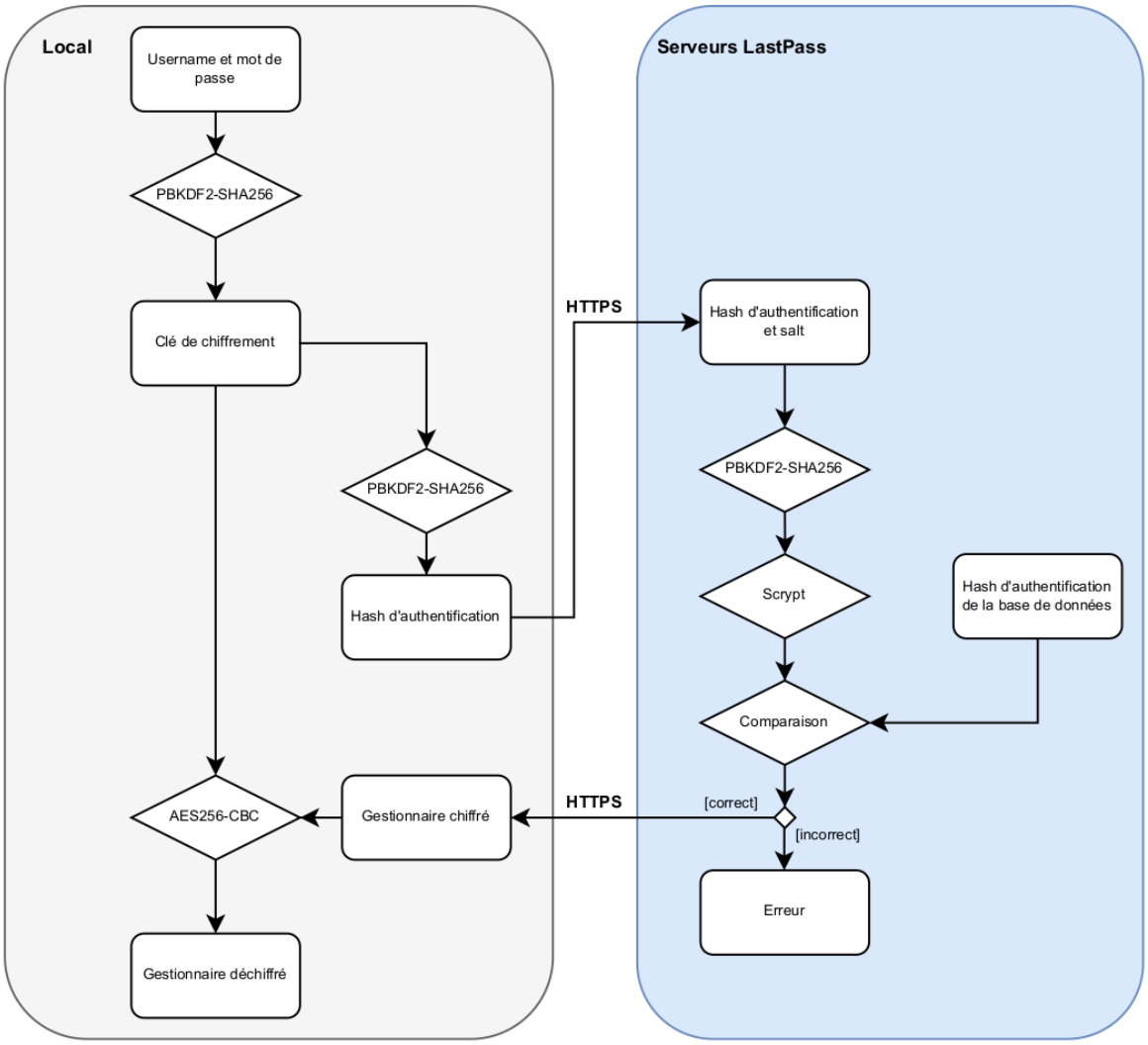

FIGURE 3.3 – Schéma de déchiffrement de LastPass

Afin de générer la clé de chiffrement, ce dernier dérive le master password en utilisant le nom d'utilisateur comme sel. Il utilise l'algorithme PBKDF2-SHA256 avec 100'100 rounds. Du côté client, l'output est hashé une fois de plus et est envoyé aux serveurs du provider afin de comparer ce hash d'authentification à celui qui est dans la base de données.

Toutes les données de l'utilisateur sont chiffrées avec AES-256-CBC avec la clé de chiffrement

générée depuis le master password.

Les serveurs de LastPass stockeront la hash d'authentification et le coffre-fort de l'utilisateur chiffré. La clé de chiffrement reste sur le device en local dans la mémoire du processus.

Tous les gestionnaires de mots de passe incluent la fonctionnalité de 2FA (voire MFA). La façon dont est géré cette sécurité additionnelle dépend du constructeur et de la méthode utilisée, certains stockent une clé supplémentaire sur les serveurs, d'autres stockent en local les informations (notammant avec les facteurs biométriques). Lors de la génération de la clé de chiffrement, le master password et les facteurs supplémentaires sont combinés pour y dériver la clé.

3.2 Partage d'informations

La majorité des gestionnaires de mots de passe proposent le partage d'informations entre plusieurs utilisateurs. Nous n'allons pas entrer dans tous les détails dans cette sous-section afin de rester bref et d'expliquer le schéma général du partage de données.

Pour cette fonctionnalité, nous utilisons la cryptographie asymétrique avec RSA. Dès l'inscription et la création du coffre-fort de l'utilisateur, une paire de clé de 2048 bits [publique, privée] est générée.

La façon dont est géré le stockage des clés dépend du constructeur mais en général, la clé publique est envoyée aux serveurs, tandis que la clé privée est soit stocké avec les données personnelles de l'utilisateur, soit envoyée aux serveurs. Cette dernière est chiffrée afin de garantir sa protection et étant donné qu'elle est personnelle et n'est pas censé être transmise entre utilisateurs, cela est tout à fait concevable. Pour le chiffrement, soit la même clé pour le chiffrement des données est utilisée ou une clé symétrique supplémentaire est générée.

Afin d'illustrer concrètement le processus complet, nous allons nous baser sur le gestionnaire Dashlane et nous montrerons un exemple où Alice souhaiterait partager un identifiant à un autre utilisateur, Bob.

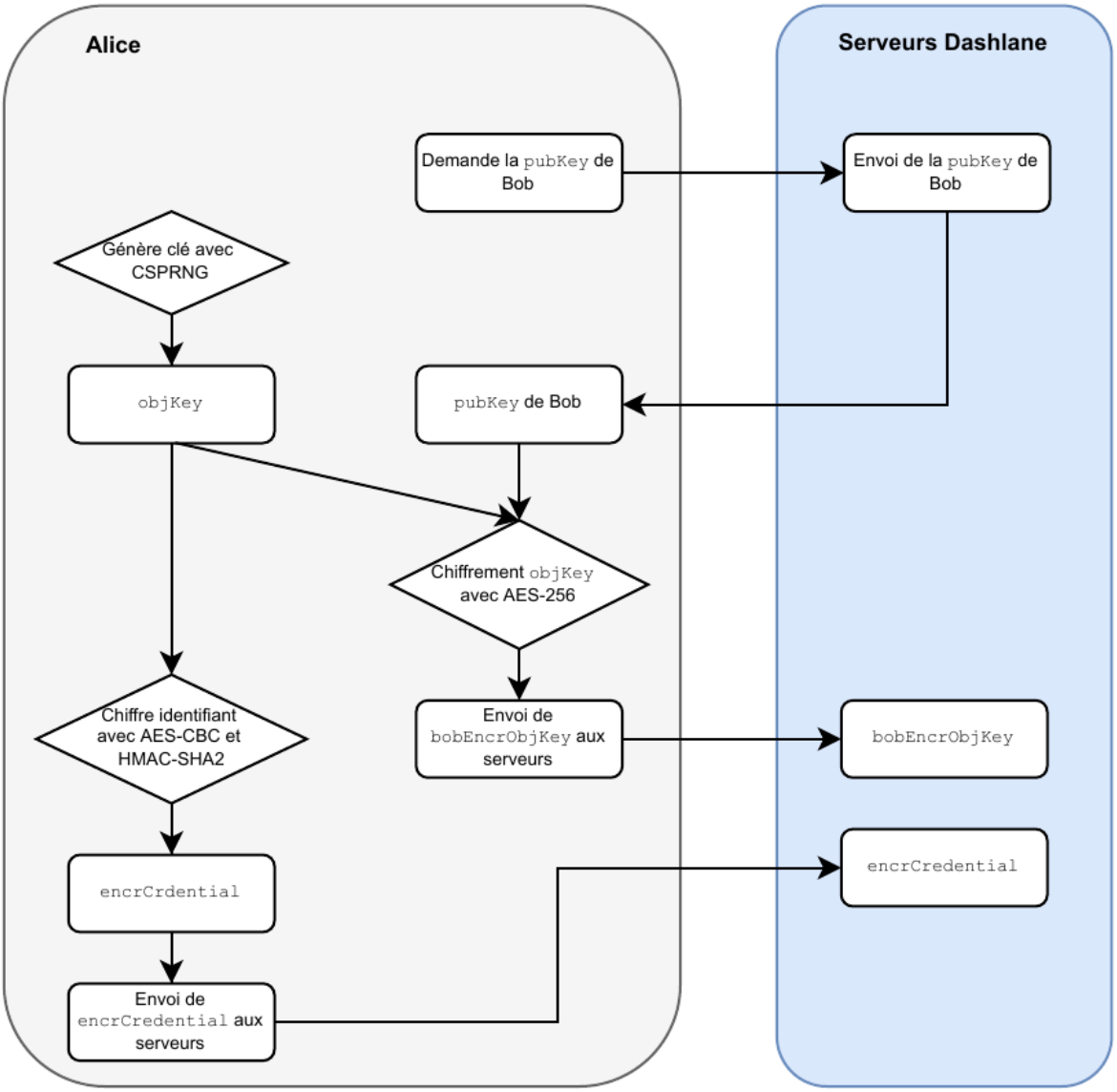

FIGURE 3.4 – Partage d'un identifiant sur Dashlane

Les différents éléments à considérer :

- `pubKey` est la clé publique de Bob et va servir à chiffrer l'information à partager
- `privKey` est la clé privée de Bob
- `objKey` est la clé symétrique AES-56 générée avec un CSPRNG qui chiffre l'identifiant à partager
- `bobEncrObjKey` est la clé unique de l'objet chiffrée avec la clé publique de Bob

- `encrCredential` est l'identifiant à partager chiffré avec la clé générée à l'étape précédente

A chaque fois qu'un élément doit être partagé, on génère une clé symétrique aléatoirement qui est stockée sur les serveurs du provider à l'aide de la clé publique de l'utilisateur concerné. La clé est générée à l'aide d'un CSPRNG. Elle va permettre de chiffrer les identifiants avec AES-CBC et HMAC-SHA2, qui va permettre de garantir la confidentialité ainsi que l'intégrité en l'authentifiant.

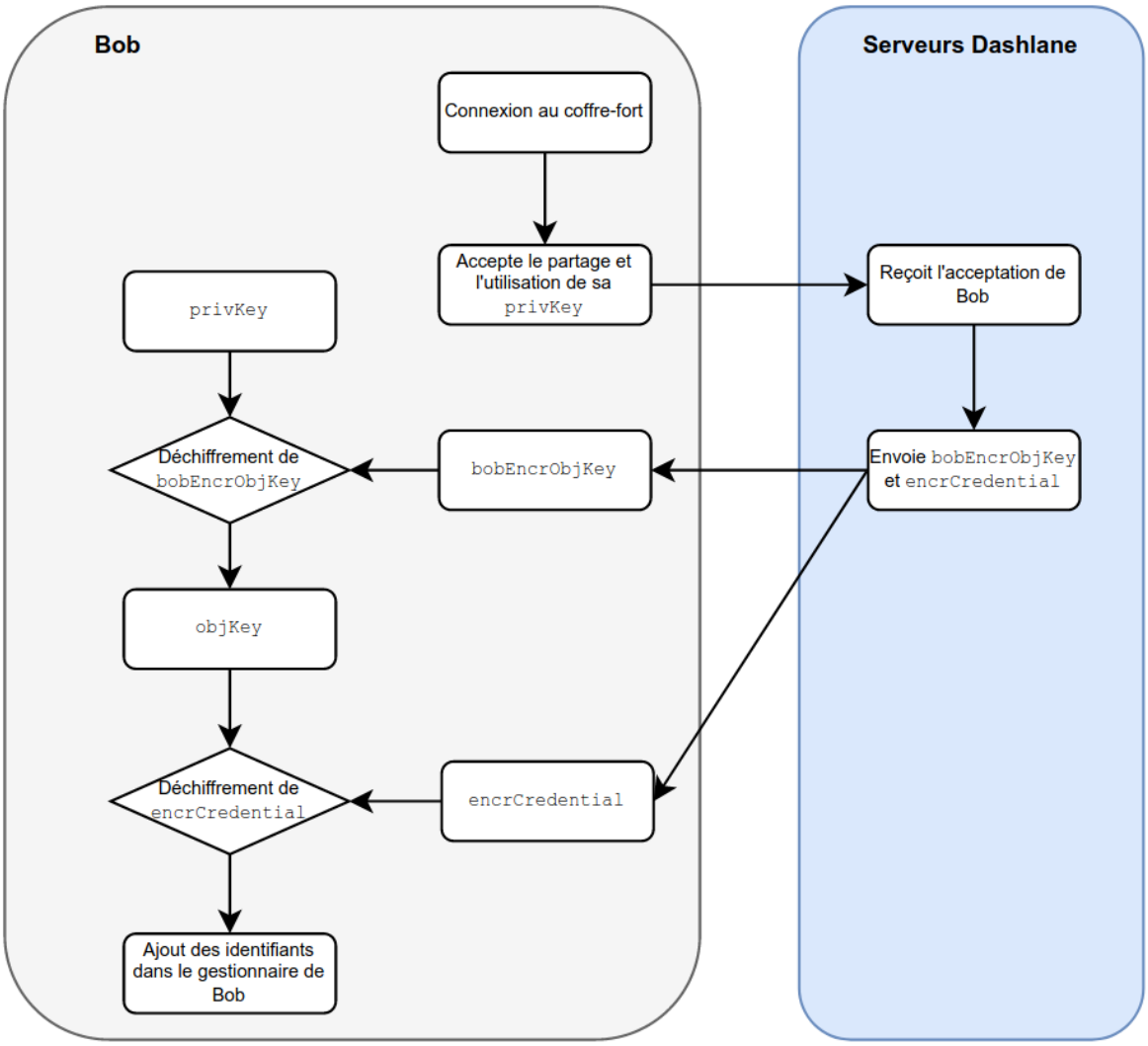

FIGURE 3.5 – Réception d'un partage d'identifiant sur Dashlane

Afin de récupérer le partage, Dashlane demande à l'utilisateur d'accepter l'envoi d'Alice pour

qu'il accepte d'utiliser sa clé privée. Tout dépend de l'architecture de l'application, mais cette dernière doit d'abord être déchiffrée avant d'être utilisée pour déchiffrer l'identifiant envoyé par Alice.

Certains gestionnaires proposent de faire des répertoires partagés. Dans le cadre d'entreprises, cela est une fonctionnalité très utile. Le processus est le même que pour le partage avec une personne. Une clé pour le dossier à partager est générée et on chiffre chaque clé publique de tous les utilisateurs concernés à l'aide de cette dernière.

3.3 3 états du gestionnaire de mot de passe

Les gestionnaires de mots de passe ont généralement 3 états différents, *Not Running*, *Unlocked State* et *Locked State*. Nous allons expliquer plus en détails comment ils fonctionnent lorsqu'ils sont dans les différents états. Nous nous basons sur un article qui analyse le management des secrets[11].

3.3.1 Etat *Not Running*

C'est l'état du password manager lorsqu'il n'a jamais été utilisé et configuré après son installation. On définit également cet état s'il n'a pas été lancé depuis le dernier redémarrage du système ou a été arrêté par un utilisateur. Dans cet état, le gestionnaire doit garantir qu'il n'y a aucune donnée sensible stockée sur le disque, comme une clé de chiffrement ou le master password.

3.3.2 Etat *Unlocked State*

Cet état indique que le gestionnaire fonctionne et donc, l'utilisateur a entré son master password afin de déchiffrer toutes les données afin d'avoir accès aux informations stockées. Le gestionnaire doit garantir qu'il n'est pas possible d'extraire aucune information sensible de la mémoire.

3.3.3 Etat *Locked State*

Nous considérons cet état lorsque l'utilisateur a lancé le gestionnaire (déjà configuré) sans avoir encore entré le master password ou qu'il a lui même verrouiller son gestionnaire. À ce moment, il ne devrait pas y avoir de données sensibles stockées sur le disque afin d'éviter toute extraction.

3.4 Algorithmes cryptographiques

Afin d'avoir une bonne vue d'ensemble de l'architecure sécuritaire des gestionnaires de mots de passe sélectionnés pour notre étude, nous allons résumer les algorithmes cryptographiques utilisés pour le chiffrement des données, la dérivation des clés et l'authentification.

	Chiffrement des données	Dérivation des clés	
LastPass	AES-CBC 256	Argon2d	PBKDF2-SHA2 100'100 rounds
Dashlane	AES-256	Argon2d	PBKDF2-SHA2 200'000 rounds
1Password	AES-GCM 256	PBKDF2-SHA2 100'000 rounds	
KeePass	AES-CBC 256 ou ChaCha20	AES-KDF	Argon2
Bitwarden	AES-CBC 256	PBKDF2-SHA2 100'000 rounds	
NordPass	XChaCha20-Poly1305-IETF	Argon2id	
Padloc	AES-GCM 256	PBKDF2-SHA2	
Keeper	AES-256	PBKDF2	
Firefox	AES-CBC ou 3DES-CBC	PBKDF2-SHA2 et HKDF 1000 rounds	

TABLE 3.1 – Algorithmes cryptographiques utilisés dans les gestionnaires

Nous pouvons remarquer que certains gestionnaire ont plusieurs algorithmes différents pour le même élément, ceci est dû au fait que l'utilisateur a le choix de configurer celui qu'il préfère et que l'application supporte les deux.

Chapitre 4

Analyse de menaces

Ce chapitre a pour but d'identifier et analyser toutes les menaces existantes et / ou potentielles des *password managers* en les modélisant en suivant un certain processus afin d'avoir une meilleure vision des risques.

Puis, nous allons rédiger toutes les exigences sécuritaires que doivent respecter les gestionnaires de mots de passe afin que ces dernières garantissent une utilisation sûre qui évite des pertes ou vol de données.

4.1 Modélisation de menaces

Afin de modéliser correctement les menaces, nous allons suivre la norme ISO 27005[12]. Cela va nous permettre de séparer la modélisation en un processus avec plusieurs étapes comme suit :

1. Établissement du contexte, qui inclut
    - Objectifs des gestionnaires de mots de passe
    - Hypothèses et exigences de sécurité
    - Actifs à haute valeur
    - Data Flow Diagram
    - Définition des critères d'analyse
2. Identification des risques
    - Identification des biens
    - Identification des menaces
    - Identification des contrôles

- Identification des vulnérabilités
- Identification des conséquences

3. Analyse des risques
4. Évaluation des risques
5. Traitement des risques
6. Documentation

La dernière étape ne sera pas explicitement abordée car elle vise à documenter le modèle de menaces que nous allons établir.

Pour chaque étape du processus, nous allons aborder les 3 types de gestionnaires existants ; cloud-based, browser-based et local-based. On peut cependant les grouper en deux catégories différentes car le cloud-based et browser-based fonctionnent de la même manière lors de synchronisations de données.

Nous allons également nous baser sur le processus de modélisation de menaces de OWASP[20].

4.1.1 Établissement du contexte

Dans cette section, nous allons comprendre les applications et comment interagissent les gestionnaires de mots de passe avec les entités externes. Au final, nous allons établir un *Data Flow Diagram* (DFD) qui va nous permettre de présenter tous les chemins différents du système en mettant en avant les vulnérabilités potentielles.

Comme cité plusieurs fois, l'objectif principal d'un gestionnaire de mots de passe est de stocker de manière **sûre** des informations sensibles, le plus souvent des identifiants, mais également la possibilité de stocker des notes, des informations bancaires, contrats, etc. Ils offrent également la fonctionnalité de générer des mots de passe fort et d'auto-compléter les champs de connexion.

On peut émettre plusieurs hypothèses de sécurité afin de mettre en avant ce que doit assurer le gestionnaire de mots de passe pour être considéré comme sûr. Nous séparons ces hypothèses en deux groupes ; un concernant l'utilisateur et un concernant l'application en elle-même.

Utilisateur

- Master password fort
- Master password unique et différent d'autres identifiants de l'utilisateur

Système

- Données du gestionnaire chiffrées / protégées
- Clés ou master password protégés dans la mémoire du processus ou complète
- Presse-papier effacé après un certain temps

- Utilisation d'algorithmes cryptographiques forts et recommandés
- Génération de mots de passe suffisamment forts
- Authentification des données
- Données effacées du disque lorsque le gestionnaire est dans un état *Not Running* ou *Locked*
- Base de données en local protégée
- Base de données des clés protégée (browser)
- Communication avec les serveurs chiffrée (cloud)
- Informations sensibles non-transmises en clair aux serveurs (cloud)
- Serveurs de confiance (cloud)

Nous pouvons à présent définir les actifs (*assets*) des gestionnaires de mots de passe, qui représentent les éléments qui ont de la valeur et qui demandent une importante protection.

Tout d'abord, un actif à haute valeur est l'**application** du gestionnaire de mots de passe. Cette dernière peut être sous forme d'application desktop, extension de navigateur ou application mobile. Nous voulons garantir :

- Disponibilité
- Intégrité

Ensuite, un autre bien important est **la base de données** qui contient tous les identifiants, notes, informations bancaires, stockées par l'utilisateur. Le bien principal sont donc les données. Nous pouvons également inclure le processus de chiffrement du coffre-fort comme bien principal. Nous souhaitons garantir les éléments suivants :

- Confidentialité des données
- Intégrité des données

Un autre actif à haute valeur sont les **clés de chiffrement** (ou les clés privées dans le cadre de partage de données avec d'autres utilisateurs). Nous incluons également le processus de génération de clés. Nous voulons garantir ces éléments :

- Confidentialité
- Intégrité

Finalement, dans le cadre de gestionnaires de mots de passe cloud-based (ou browser-based) qui ne fonctionnent pas en mode offline, nous voulons protéger les données des **serveurs** du provider. Le processus de transfert de données est également important car les données doivent impérativement être protégées. Ainsi, nous souhaitons assurer :

- Confidentialité
- Disponibilité du service

À présent, nous allons définir le DFD afin de correctement décomposer le système. Pour avoir la meilleure vue d'ensemble, nous allons faire un diagramme pour les gestionnaires local-based et un autre pour les cloud-based / browser-based.

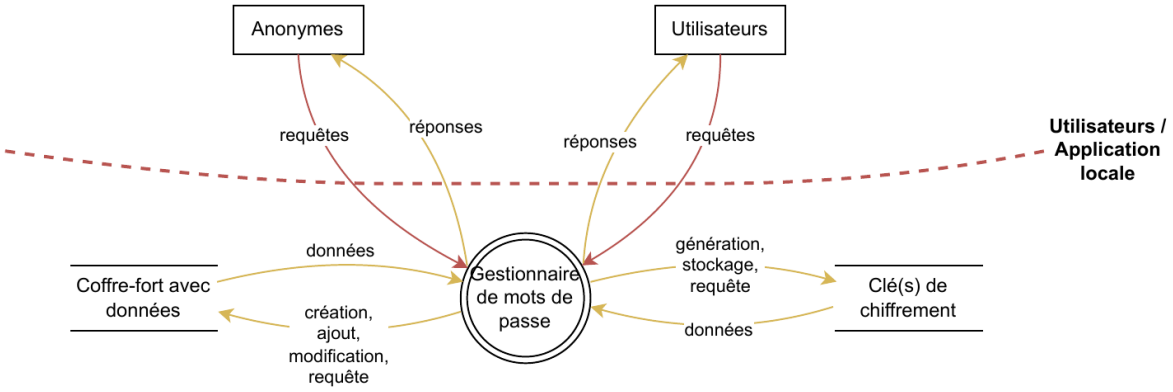

FIGURE 4.1 – Data Flow Diagram pour les gestionnaires local-based

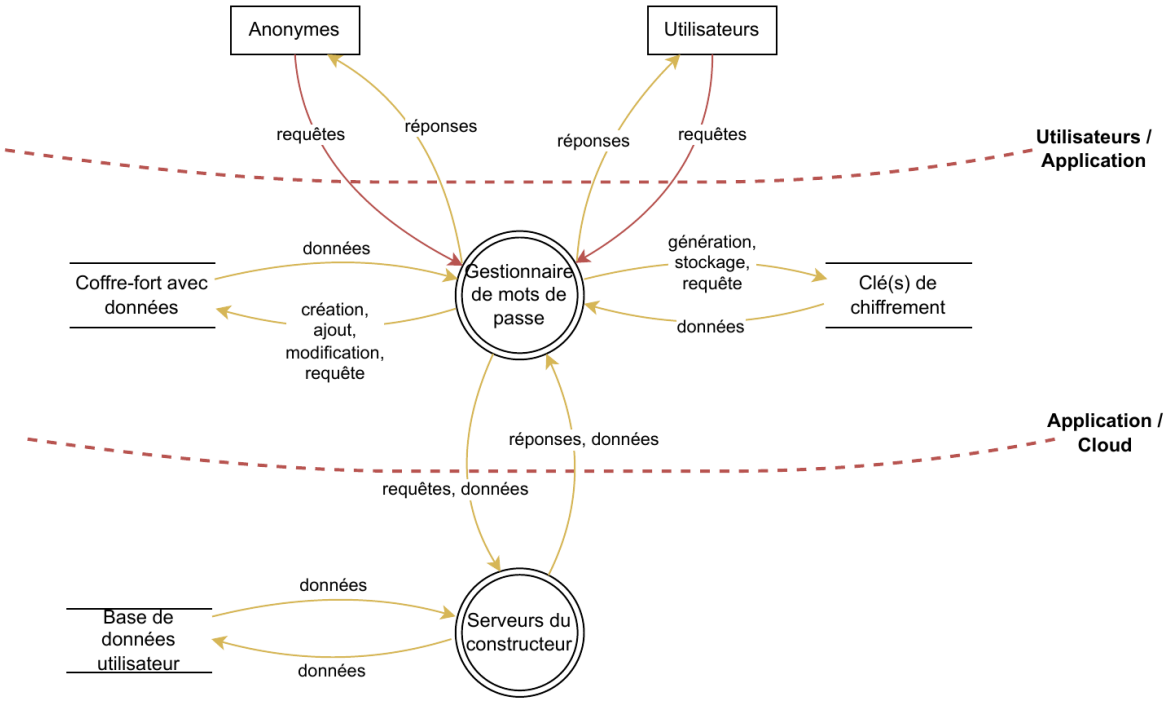

FIGURE 4.2 – Data Flow Diagram pour les gestionnaires cloud-based et browser-based

Dernièrement, nous allons définir les différents critères et niveaux pour évaluer l'impact des

événements (les conséquences), la probabilité d'événement, la sévérité des vulnérabilités ainsi que l'évaluation des risques. Ces niveaux seront réutilisés pour l'identification des risques dans la suite de l'analyse de menaces.

Échelle	Description	Valeur
Aucune	La vulnérabilité ne présente aucune sévérité	0
Bas	La vulnérabilité a peu d'impact sur l'entreprise et demande un accès physique ou local au système	0.1-39
Moyen	En général, elle demande à l'attaquant d'être sur le même réseau que la victime et elle demande les privilèges utilisateurs pour effectuer une exploitation	4.0-6.9
Haut	La vulnérabilité est difficile à exploter, peut être une élévation de privilèges et peut amener à une importante perte de données ou de temps d'arrêt	7.0-8.9
Critique	Exploitation de vulnérabilités au niveau root, l'attaquant n'a pas besoin d'être authentifié ou avoir des connaissances sur la victime	9.0-10.0

Table 4.1 – Scores de sévérité basés sur CVSS v3.1

Échelle	Description	Valeur
Négligeable	L'impact sur l'entreprise est négligeable	0
Mineur	L'effet sur les biens de l'entreprise est limité, comme entraîner des pertes financières mineures ou entraîner des dommages mineures aux actifs de l'entreprises	1
Modéré	L'effet sur les biens peut être grave, les effets causés sur les biens seront considérés comme importants	2
Important	L'effet sur les biens de l'entreprise peut être grave à catastrophique	3
Catastrophique	On s'attend à ce que la menace ait de multiples effets graves à catastrophiques sur les biens de l'entreprise	4

Table 4.2 – Impact des menaces sur l'entreprise

Échelle	Description	Valeur
Rare	La probabilité que l'événement arrive est rare	0
Peu probable	La probabilité que l'événement arrive est peu probable	1
Possible	La probabilité que l'événement arrive est possible	2
Probable	La probabilité que l'événement arrive est probable	3
Certain	La probabilité que l'événement arrive est certain	4

Table 4.3 – Probabilités d'événements

Pour l'évaluation des risques, il est important de prendre en considération la probabilité d'événements et les conséquences sur l'entreprise afin de définir une échelle d'évaluation de risques. Nous utilisons une matrice de risques basée sur une étude quantitative.

FIGURE 4.3 – Critères d'évaluation des risques

4.1.2 Identification des risques

Cette section va nous servir à déterminer ce qui peut se produire pour causer une perte potentielle et à comprendre comment, où et pourquoi la perte peut se produire.

Afin d'établir une étude complète, nous allons premièrement identifier les biens, menaces (avec les sources de menaces et les scénarios), les contrôles, les vulnérabilités ainsi que les conséquences sur l'entreprise.

Dans chaque table, pour chaque élément cité, nous précisons les types de gestionnaires de mots de passe concernés ; L (local-based), B (browser-based), C (cloud-based).

Identification des biens

Nous avons déjà identifié les actifs plus-haut dans l'analyse, cependant nous allons ajouter un code pour les identifier et les référencer plus tard dans l'identification des risques

Code	Bien	Description	Type
A1	Application	L'application du gestionnaire de mots de passe	L, B, C
A2	Base de données	Base de données avec toutes les informations stockées	L, B, C
A3	Clés de chiffrement	Clés de chiffrement ou clés pour le partage de données	L, B, C
A4	Serveurs du provider	Serveurs dans le cloud où sont stockés les données de l'utilisateur	B, C

TABLE 4.4 – Biens des gestionnaires de mots de passe

Identification des menaces

Nous allons identifier toutes les sources de menaces possibles des gestionnaires de mots de passe avec leurs motivations (uniquement si c'est de source humaine) afin de pouvoir mettre en avant les différents scénarios de menaces.

Code	Source de menace	Motivations
S1	Hackers	Amusement, gloire, challenge, argent, ego
S2	Script kiddies	Curiosité, ego, argent, amusement
S3	Concurrent	Espionnage, réutilisation de contenu
S4	Cybercrime	Argent, destruction de données
S5	Accident humain	-
S6	Problèmes techniques	-

TABLE 4.5 – Sources de menaces des gestionnaires de mots de passe

Ci-dessous, se trouve un tableau avec différents scénarios de menaces par rapport aux biens que nous avons défini précédemment. Nous nous basons sur le modèle STRIDE afin de définir les types de menaces.

	Code	Code du bien	Scénario de menace	Type de menace	Source de menace	Type
1	T1	A1	Brute-force sur le master password	*Spoofing*	S1, S2, S4, S5	L, B, C
2	T2	A1	Key-logger installé sur le device de l'utilisateur et qui sniff le master password	*Elevation of privileges*	S1, S2, S4	L, B, C
3	T2	A1	Utilisateur non-déconnecté de son compte et attaquant a accès au device	*Elevation of privileges*	S1, S2, S4, S5	L, B, C
4	T1	A1	Mauvais mécanisme d'authentification	*Spoofing*	S1, S2, S4, S6	L, B, C
5	T3	A1	Lecture du presse-papier	*Information disclosure*	S1, S2, S4	L, B, C
6	T1	A1	Phishing en imitant une page de connexion	*Spoofing*	S1, S2, S3, S4	B, C
7	T1	A1	Attaque XSS avec les champs de connexion auto-complétés	*Spoofing*	S1, S2, S4	B, C
8	T1	A1	Coffre-fort compromis dû à des identifiants volés sur d'autres sites	*Spoofing*	S1, S2, S4, S5	L, B, C
9	T3	A1	Récupération de données sensibles en clair sur le disque	*Information disclosure*	S1, S2, S4	L, B, C
10	T3	A1	Clickjacking	*Information Disclosure*	S1, S2, S4	B, C
11	T3	A2	Récupération de la base de données et déchiffrement dû à un algorithme trop simple	*Information disclosure*	S1, S2, S4	L, B, C
12	T3	A2	Récupération de données sensibles en mémoire	*Information Disclosure*	S1, S2, S4	L, B, C
13	T6	A2	Injection SQL dans le base de données utilisateur	*Tampering*	S1, S2, S4	L, B, C
14	T3	A3	Récupération de la clé de chiffrement en mémoire	*Information Disclosure*	S1, S2, S4	L, B, C
15	T3	A3	Brute-force de clé		S1, S2, S4	L, B, C
16	T3	A4	MITM (Interception de données non-chiffrées)	*Information Disclosure*	S1, S2, S4, S6	B, C
17	T3	A4	Vols de données non protégées sur le serveur	*Information disclosure*	S1, S2, S4, S6	
18	T4	A4	DoS	*Denial of service*	S1, S2, S4, S6	B, C
19	T5	A4	Perte des données utilisateurs dû à un serveur down	*Equipment failure*	S5, S6	B, C

A1 = Application | A2 = Base de données | A3 = Clés | A4 = Serveurs du provider

TABLE 4.6 – Scénarios et types de menaces possibles

Identification des contrôles

Une étape important lors de l'identification des risques et d'identifier les contrôles qui existent déjà sur les applications sur lesquelles nous basons notre analyse de risques. Étant donné, que notre étude est plutôt générale car elle ne se base pas sur un seul gestionnaires de mots de passe, nous allons énumérer les contrôles qui ont été entrepris afin de baisser le niveau de risque. Nous parlerons en détails des contrôles effectués lors de la seconde partie du travail.

Le contrôle qui est présent sur toutes les applications est que chaque utilisateur possède une clé de chiffrement différente, qui est dérivée avec son master password. Même si ce dernier est le même, la clé ne sera pas pareille dû au salt qui est soit aléatoire soit le username, qui est unique. Les algorithmes choisis sont forts (voir 3.1) et sont recommandés pour la dérivation des clés. Ainsi, la clé est protégée contre le brute-force ou les attaques par dictionnaire. Néanmoins, il est quand même important d'avoir un master password fort [1], ce qui n'est pas demandé dans tous les gestionnaires de mots de passe lors de l'inscription de l'utilisateur.

Au niveau du chiffrement des données, les gestionnaires local-based et cloud-based, utilisent AES-256. L'algorithme est également encore recommandé en 2022.

Au niveau de la protection mémoire, certains proposent une protection des données sensibles dans le processus (avec DPAPI ou Chacha20) et les données sont effacées sur le disque dès que le processus s'arrête. En se basant sur une étude[11], la mémoire est en général bien protégée lorsque le gestionnaire dans l'état *not running*, cependant dès le moment où le gestionnaire est dans l'état *unlock* ou *locked*, les secrets et le master password sont plus exposés.

Au niveau de la protection du keylogger, aucune protection n'est réellement implémentée, à part utiliser l'application dans environnement sécurisé comme *Secure Desktop* sur Windows. Pour la protection du presse-papier, certaines applications (comme 1Password ou Keepass) mettent à disposition une fonctionnalité pour enlever les données sensibles du presse-papier après un certain moment.

Finalement, pour l'auto-complétion des champs de connexion, certains s'assurent que l'utilisateur valide s'il veut vraiment remplir ce champ afin d'éviter d'entrer les données sensibles dans des pages web clonées malveillantes.

Identification des vulnérabilités

Pour chaque actif identifié au préalable, nous allons analyser les vulnérabilités qui pourraient être présentes. Nous allons également ajouter la sévérité de la vulnérabilité.

1. Ce qu'on considère un mot de passe fort ; au moins une majuscule, une minuscule, un chiffre et 8 caractères

Bien	Vulnérabilité	Sévérité de la vulnérabilité	Type
A1	Master password trop faible	Haut	L, B, C
A1	Temps de session long	Critique	L, C
A1	Presse-papier non nettoyé après un certain temps	Moyen	L, B, C
A1	Master password non unique	Bas	L, B, C
A1	2FA pas proposé par défaut	Haut	L, B, C
A1	Aucune protection de la mémoire du processus	Haut	L, B, C
A1	Aucune authentification effectuée	Moyen	L, B, C
A1	Option *remember me*	Haut	L, C
A1	Génération de mots de passe faibles	Haut	L, B, C
A1	Auto-complétion de champs sans une interaction avec l'utilisateur	Moyen	B, C
A1	Utilisation de critères de correspondance faibles lors de l'auto-complétion	Moyen	B, C
A1	Données non nettoyées sur le disque lors de l'arrêt du processus	Haut	L, B, C
A1	Manque de contrôle des entrées utilisateurs	Haut	L, B, C
A1	Règles d'auto-complétion HTTP ne différencie pas HTTP et HTTPS	Haut	B, C
A2, A3	Utilisation d'algorithmes de chiffrement plus recommandés ou trop faibles	Haut	L, B, C
A2, A4	Informations stockées en clair	Haut	L, B, C
A3	Dérivation des clés trop simple	Moyen	L, C
A4	Communication non-chiffrée	Critique	B, C
A4	Backup non effectué	Critique	B, C

A1 = Application | A2 = Base de données | A3 = Clés | A4 = Serveurs du provider

TABLE 4.7 – Vulnérabilités présentes dans les gestionnaires de mots de passe

Identifications des conséquences

Il est également important d'identifier les conséquences qui pourraient être causés par un incident. Nous allons définir par bien actif quelles sont les conséquences qu'ils pourraient y avoir sur l'entreprise lors d'une perte de confidentialité, intégrité ou disponibilité. Pour cela, nous allons nous référer au tableau 4.6 qui définit différents scénarios de menaces.

Bien	Conséquences
Application (A1)	Perte de réputation et d'image
Base de données (A2)	Perte de réputation et d'image, perte de données
Clés de chiffrement (A3)	Perte de réputation et d'image, perte de données
Serveurs (A4)	Perte de réputation et d'image, coûts financiers, coûts de réparation

TABLE 4.8 – Conséquences des menaces sur l'entreprise

4.1.3 Analyse des risques

L'objectif de cette section est d'estimer la probabilité des incidents et les conséquences pour les biens qu'ils menacent. Pour ce faire, nous allons faire une analyse de risque qualitative ce qui va nous permettre de faire une première analyse de risques assez générale afin de nous permettre d'aller plus en détails dans la seconde partie du travail de Bachelor.

Ainsi, nous allons analyser l'impact des conséquences ainsi que la probabilité que chaque scénario d'attaque identifié puisse se produire. Pour cela, nous allons utiliser plusieurs notations différentes ; pour les conséquences nous aurons les niveaux suivants :

Bien	Menace	Scénario	Impact des conséquences	Probabilité	Niveau de risque
A1	T1	Brute-force sur le master password	Important	Certain	Haut
A1	T2	Key-logger installé sur le device de l'utilisateur et qui sniff le master password	Important	Possible	Moyen
A1	T2	Utilisateur non-déconnecté de son compte et attaquant a accès au device	Important	Probable	Haut
A1	T1	Mauvais mécanisme d'authentification	Modéré	Rare	Bas
A1	T3	Lecture du presse-papier	Modéré	Probable	Moyen
A1	T1	Phishing en imitant une page de connexion	Modéré	Possible	Moyen
A1	T1	Attaque XSS avec les champs de connexion auto-complétés	Important	Possible	Moyen
A1	T1	Coffre-fort compromis dû à des identifiants volés sur d'autres sites	Mineur	Possible	Moyen
A1	T3	Clickjacking	Modéré	Possible	Moyen
A1	T3	Récupération de données sensibles en clair sur le disque	Important	Probable	Haut
A2	T3	Récupération de la base de données et déchiffrement dû à un algorithme trop simple	Important	Peu probable	Moyen
A2	T3	Récupération de données sensibles en mémoire	Important	Probable	Haut
A2	T6	Injection SQL dans la base de données	Important	Peu probable	Moyen
A3	T3	Récupération de la clé de chiffrement en mémoire	Important	Probable	Haut
A3	T3	Brute-force de clé	Important	Peu probable	Moyen
A4	T3	MITM (Interception de données non-chiffrées)	Catastrophique	Peu probable	Moyen
A4	T3	Vols de données non protégées sur le serveur	Important	Rare	Moyen
A4	T4	DoS	Modéré	Possible	Moyen
A4	T5	Perte des données utilisateurs dû à un serveur down	Catastrophique	Peu probable	Moyen

A1 = Application | A2 = Base de données | A3 = Clés | A4 = Serveurs du provider
T1 = Spoofing | T2 = Elevation of privileges | T3 = Information Disclosure | T4 = Denial of service | T5 = Equipment failure | T6 = Tampering

TABLE 4.9 – Analyse des risques de chaque scénario de menace

4.1.4 Évaluation des risques

Cette étape sert à évaluer si les risques sont acceptables ou s'ils ont besoin d'un traitement supplémentaires, c'est-à-dire une mitigation. Nous avons défini à l'étape précédente le niveau de risque de chaque scénario de menace. En résumé, nous avons :

- Bas : 1
- Moyen : 13
- Haut : 5

Ainsi, nous pouvons définir que les tous les scénarios de menaces ayant un niveau de risque

bas peuvent être acceptés par l'entreprise, néanmoins ces derniers ne doivent pas être oubliés et mis de côté. Pour les niveaux de risques moyen et haut, il est nécessaire d'appliquer une mitigation et de surveiller les menaces.

4.1.5 Traitement des risques

Dans cette section, nous allons définir les contre-mesures à entreprendre afin d'éviter au maximum les menaces que nous avons identifié plus haut. Comme précisé dans le paragraphe de l'identification des contrôles 4.1.2, beaucoup de choses sont déjà mises en place par les constructeurs et sont efficaces contre les attaques. Néanmoins, nous pouvons ajouter quelques contre-mesures qui ne sont pas ou peu effectuées par les quelques candidats que nous avons sélectionnés au préalable. Ainsi, nous allons sélectionner les risques qui ont été qualifiés comme moyen ou haut et nous allons proposer des contre-mesures possibles.

Menace	Contre-mesures
Master password faible	- Ajouter des contrôles lors de l'inscription de l'utilisateur en lui demandant un mot de passe fort - Proposer un master password fort généré aléatoirement par l'application
Temps de session trop long	- Ajouter un temps d'expiration de session et verrouiller le gestionnaire après ce temps
Presse-papier non nettoyé	- Après un temps court, nettoyer le presse-papier de l'utilisateur
Phishing en clonant une page de connexion	- Appliquer un meilleur critère de match lors de l'auto-complétion des champs de connexion - Demander à l'utilisateur de valider l'auto-complétion
Clickjacking	- Bloquer l'utilisation de Javascript à l'aide d'API sécurisées
Communication non-chiffrée	- Ajouter HTTPS pour la communication avec le serveur - Éviter d'envoyer le master password ou des clés de chiffrement ou privée au serveur - Chiffrer toutes les données avant de les envoyer dans le cloud
Dérivation des clés trop simple	- Utiliser un algorithme recommandé et fort (PBKDF2-SHA2 par exemple) - Une clé de chiffrement unique par utilisateur pour éviter le brute force
Perte de données	- Si cloud-based / browser-based, effectuer des backups sur les serveurs chaque nuit - Mettre à disposition la fonctionnalité de backup pour les particuliers pour les faire en local - Exportation des données CSV et les stocker dans un endroit chiffrer et sécurisé
Manque de protection dans la mémoire	- Nettoyage de mémoire lorsque le gestionnaire est dans l'état locked - Protection des données sensible avec des APIs pour limiter l'exposition de secrets
Option remember me	- Désactivation de mode offline avec les gestionnaires cloud-based et browser-based

TABLE 4.10 – Contre-mesures possibles pour les gestionnaires

4.2 Exigences sécuritaires à respecter

Dans cette dernière section de l'analyse de menaces, nous allons établir une liste d'exigences sécuritaires que les gestionnaires de mots de passe doivent respecter afin de se définir sécurisé et afin de protéger les données sensibles au maximum. Pour cela, nous allons lister les exigences générales ainsi que les exigences par type d'applications (cloud, browser et local).

Chiffrement des données

Le chiffrement du coffre-fort qui contient toutes les données personnelles et sensibles de l'utilisateur doit impérativement être chiffré avec un algorithme ainsi qu'une taille de clé qui sont recommandés. Nous pouvons nous baser sur les recommandations du report de ECRYPT[7]. Au mieux, l'algorithme devrait être encore recommandé pour une utilisation future.

Pour les gestionnaires cloud-based et browser-based, dès le moment où il y a une interaction avec les serveurs du constructeur, le chiffrement devrait être *end-to-end* pour assurer la protection des données lors du transport dans le cloud.

Clés de chiffrement

L'algorithme utilisé pour la génération de clé.s en dérivant le mot de passe devrait se baser sur des algorithmes *Password-Based Key Derivation* et s'assurer qu'il fasse partie des recommandations d'ECRYPT. Les plus couramment utilisés sont Argon2 ou PBKDF2 (associé à une fonction de hash). Pour le salt, il doit absolument être unique. Le mieux serait d'utiliser un pseudo-nombre généré aléatoirement (PRNG) afin de s'assurer que ce nombre soit correctement aléatoire. Ce dernier devrait être stocké de manière sûre en local chez l'utilisateur. Une autre manière de faire, plus simple pour la gestion du salt aléatoire, est de prendre le username de l'utilisateur, qui peut être soit une adresse e-mail soit le nom d'utilisateur. Cependant, lors de l'inscription, il faut absolument que ce dernier soit unique afin d'éviter que des coffre-forts puissent avoir la même clé.

Pour le cloud-based et browser-based, la clé ne doit jamais être envoyée aux serveurs et doit rester en local dans la mémoire du processus.

Authentification des données et de l'utilisateur

Pour les gestionnaires local-based / offline, étant donné qu'il n'y a aucune interaction avec un serveur externe, il est nécessaire d'authentifier les données afin d'assurer l'intégrité et la confidentialité. Il est bien d'utiliser un schéma *Encrypt-then-MAC* qui est le chiffrement authentifié le plus sûr.

Pour les gestionnaires cloud-based ou browser-based, il est nécessaire d'authentifier les utilisateurs auprès des serveurs du constructeur. Pour cela, il existe plusieurs manières de faire et nous allons pas aller en détails dans cette section, mais le hash d'authentification doit être envoyé aux serveurs de manière sécurisée, c'est-à-dire que la communication doit être chiffrée avec HTTPS. Le hash d'authentification à être comparé avec celui envoyé par l'utilisateur, doit être stocké dans la base de données de l'utilisateur sur les serveurs, chiffré pour garder sa confidentialité.

Master password

Au mieux, il serait un bon réflexe, d'ajouter des exigences sécuritaires au niveau du master password lors de l'inscription de l'utilisateur. Ainsi, on peut s'assurer que le master password est fort et on peut éviter au maximum les attaques de brute-force ou par dictionnaires.

Le master password ne devrait jamais être stocké en mémoire afin d'éviter tout vol. Ainsi, la fonctionnalité de se souvenir du master password est fortement à déconseiller, à moins que des précautions de protection de mémoire aient été prises.

Une bonne pratique serait d'activer le 2FA ou MFA obligatoire par défaut afin d'ajouter une couche sécuritaire.

Pour les applications cloud-based ou browser-based, les serveurs ne devraient pas avoir connaissance du master password et il ne devrait jamais être transmis.

Session

Pour éviter que si une personne malveillante puisse accéder à votre device et exploite le gestionnaire de mots de passe, il est nécessaire d'ajouter un temps de session. Après ce court temps, l'application devrait se mettre dans un état *Locked* pour déconnecter l'utilisateur du coffre-fort.

Mémoire

Comme on l'a expliqué précédemment (3.3), chaque état différent de la mémoire doit absolument garantir différents éléments. Dans l'état *Not Running*, aucune donnée sensible ne doit rester sur le disque ; tout doit être nettoyé lors de l'arrêt du processus. Dans l'état *Unlocked State*, on doit garantir qu'il n'est pas possible d'extraire de données sensibles en mémoire (en faisant un dump de mémoire par exemple). Finalement, dans l'état *Locked*, toutes les données doivent être effacées du disque pour éviter quelconque extraction.

Presse-papier

Il est très important d'éviter des attaques de sniffing de presse-papier, pour cela, il est nécessaire de nettoyer ce dernier après un court instant (10-40 secondes) et également s'assurer que l'historique du presse-papier est effacé.

Phishing (et autres attaques web-based) et auto-complétion

Finalement, les dernières exigences qu'on peut lister sont en rapport aux attaques Web ; phishing, clickjacking, XSS, etc. Premièrement dans le cadre d'auto-complétion dans les formulaires de login, il est nécessaire demander à l'utilisateur de valider le remplissage que propose le gestionnaire pour s'assurer qu'il n'y a aucun risque.

Puis, on peut également redéfinir les critères de matching pour les URLs afin de s'assurer qu'ils soient stricts. Un autre point important où il faut assurer une protection est pour le sous-domaine de l'URL, quelques gestionnaires ignorent le sous-domaine, ainsi il serait possible de voler les identifiants du domaine parent.

Il est nécessaire faire une différence avec HTTP et HTTPS lors du remplissage de formulaire afin de ne pas subir une attaque MITM et de remplir une page clonée HTTP d'un site initialement en HTTPS.

Finalement, il faudrait bloquer l'exécution Javascript dans le cadre de gestionnaires cloud-

based ou browser-based afin d'éviter toute exécution de code malveillant et non-souhaité.

Chapitre 5

Sélection des candidats

Dans ce chapitre, nous allons faire la sélection des gestionnaires de mots de passe que nous analyserons dans le chapitre suivant. Nous définirons les critères des sélections afin de correctement les choisir, puis nous ferons notre choix en fonction des candidats sélectionnés. Afin d'avoir une bonne vue d'ensemble sur les fonctionnalités et la sécurité de chaque application, nous allons reprendre les 9 gestionnaires analysés dans les chapitres précédents.

5.1 Critères de sélection

Cette section va ainsi présenter les critères de sélections pour les candidats avec lesquels on fera une analyse sécuritaire détaillée.

Open-source - un critère assez important car un gestionnaire de mots de passe open-source pourrait être plus sûr qu'un gestionnaire closed-source dû au fait qu'importe quel utilisateur peut auditer le code source indépendamment et peut reporter des failles aux constructeurs. Les applications closed-source comptent à 100% sur l'équipe de développement et pourrait faire face à plus d'attaques.

Types - nous allons sélectionner des gestionnaires des 3 types différents afin d'avoir un éventail complet de gestionnaires de mots de passe existants. Pour faire un rappel, les différents types sont : cloud-based, local-based et browser-based.

Gratuit - il est bien de prendre en compte ce critère afin d'analyser si un gestionnaire proposant des fonctionnalités gratuites est plus faible qu'un gestionnaire complètement payant. De plus, il pourrait être intéressant de comparer la sécurité d'un application qui propose des fonctionnalités gratuites et payantes.

Nombre d'utilisateurs - il est intéressant d'ajouter le critère de la popularité afin de se rendre compte de l'impact de quelconques vulnérabilités présentes sur l'application et de

constater malgré une grande popularité, s'il y a l'existence de failles non-corrigées ou non-prises en compte.

Partage de données - cette fonctionnalité est assez critique, dû au fait que des données sont partagées entre plusieurs utilisateurs, il est impératif que les données soient uniquement accessibles par les bonnes personnes de confiance. Ainsi, il serait intéressant de sélectionner au moins un candidat qui propose cette fonctionnalité pour évaluer la sécurité.

Choix cryptographiques - ce critère concerne les choix cryptographiques utilisés pour le chiffrement des données par exemple, ou la dérivation des clés. Il sera important d'évaluer si ces choix sont suffisant. Ainsi, nous allons sélectionner des candidats qui ont effectuer des choix cryptographiques différents.

Plateforme supportée - l'analyse sécuritaire se fera dans un premier temps sur un ordinateur avec Windows 11, donc les applications doivent être disponibles pour cet OS. La raison pour laquelle nous faisons ce choix est que c'est un des OS les plus populaires sur le marché.

Failles connues - pour quelques gestionnaires de mots de passe, des vulnérabilités ont été découvertes. Pour certaines, des corrections ont été faites aux applications mais certaines n'ont pas été corrigée malgré un report vers les constructeurs des gestionnaires. Ainsi, ceci est un aspect important à prendre en compte afin de constater si les failles ont été corrigées après leurs découvertes.

5.2 Choix

Afin d'avoir une bonne vue d'ensemble sur tous les candidats et de faire une sélection pertinente, nous allons établir un tableau récapitulatif. Lors du choix, nous allons tenter de couvrir tous les critères proposés avant, afin de faire une analyse complète et diverse sur tout ce qui existe sur le marché actuellement.

Nous allons donc choisir X candidats pour avoir une bonne vue d'ensemble sur tout ce qui existe.

	Type	Open-source?	Gratuit?	Nombre d'utilisateurs	Partage de données	Choix cryptographiques	Disponible sur Windows 11 ?	Failles ou faiblesses connues ?
Dashlane	Cloud	Non	Oui	15'000'000	Oui	AES 256 Argon2d / PBKDF2-SH2	Oui	Oui
LastPass	Cloud	Non	Oui	33'000'000	Oui	AES-CBC 256 Argon2d / PBKDF2-SH2	Oui	Oui
KeePass	Local	Oui	Oui	60'000'000	Non	AES-CBC 256 / Chacha20 AES-KDF / Argon2	Oui	Oui
1Password	Cloud	Non	Non	15'000'000	Oui	AES-GCM 256 PBKDF2-SHA2	Oui	Oui
NordPass	Cloud	Non	Oui	14'000'000	Oui	XChaCha20 Argon2id	Oui	-
Bitwarden	Cloud	Oui	Oui	15-20'000'000	Oui	AES-CBC 256 PBKDF2-SHA2	Oui	Oui
Keeper	Cloud	Non	Non	1'000'000	Oui	AES 256 PBKDF2	Oui	Oui
Padloc	Cloud	Oui	Oui	-	Oui	AES-GCM 256 PBKDF2-SHA2	Oui	Oui
Firefox	Browser	Oui	Oui	-	Non	AES-CBC / 3DES-CBC PBKDF2-SHA2 / HKDF	Oui	Oui

FIGURE 5.1 – Comparatif des candidats pour la sélection

Ainsi, nous allons sélectionner :

- LastPass
- KeePass
- Firefox
- 1Password

Chaque chapitre dédié aux gestionnaires de mots de passe sélectionnés se basera sur les exigences sécuritaires que nous avons définis au chapitre précédent et sur les vulnérabilités et faiblesses déjà connues.

À chaque début de chapitre, nous établirons les critères d'évaluation de la sécurité de l'application. Les critères seront à chaque fois différents en fonction du type ou des fonctionnalités proposées.

Chapitre 6

KeePass

6.1 Critères d'analyse

Chapitre 7

LastPass

Ce chapitre sera dédié à l'analyse sécuritaire de l'application LastPass. Nous allons utiliser l'extension de navigateur sur Google Chrome ainsi que sur Firefox. Il existe également des applications desktop, cependant les versions ne sont pas proposées sur le site officiel car ils mettent en avant uniquement l'extension de navigateur. Nous n'allons donc pas nous concentrer sur l'application desktop et uniquement analyser l'extension de navigateur.

7.1 Environnement

Logiciel	Version
Windows 11	10.0.22621
Google Chrome	106.0.5249.119
Firefox	106.0.1
LastPass Extension	4.101.1

Table 7.1 – Environnement utilisé pour LastPass

7.2 Critères d'analyse

Chapitre 8

Conclusion

Bibliographie

[1] 1Password. 1password security design, 2021.

[2] Apr4h. Decrypting browser credentials for fun (but not profit), 2019.

[3] bitwarden. Bitwarden security whitepaper, 2022.

[4] bitwarden. World password day global survey full report, 2022.

[5] Clifford Colby, Rae Hodge, and Attila Tomaschek. Best password manager to use for 2022, 2022.

[6] Dashlane. Security white paper, 2022.

[7] ECRYPT-CSA. Algorithms, key size and protocols report, 2018.

[8] Elizabeth A. Gallagher. Choosing the right password manager. *Serials Review*, 45 :1–2, 84–87, 2019.

[9] Eric Griffith. How to master google password manager, 2022.

[10] Jingxin Hong Hengwei Zhang and Jun Hu. Analysis of encryption mechanism in keepass password safe 2.30. *2016 10th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID)*, pages 43–46, 2016.

[11] ISE. Password managers : Under the hood of secrets management, 2019.

[12] Technologies de l'information — Techniques de sécurité — Gestion des risques liés à la sécurité de l'information. Standard, International Organization for Standardization, July 2018.

[13] KeePass. Security, 2022.

[14] Keeper. Keeper encryption model, 2022.

[15] Michael Kurko. Best password managers, 2022.

[16] LastPass. Technical whitepaper.

[17] lgg. Keepass file format explained, 2017.

[18] Paulius Masiliauskas. Most secure password managers in 2022, 2022.

[19] NordPass. Nordpass business whitepaper, 2022.

[20] Larry Conklin (OWASP). Threat modeling process, 2022.

[21] Padloc. Security whitepaper, 2022.

[22] PasswordManager. Password manager trust survey, 2020.

[23] Tom Ritter. Private by design : How we built firefox sync, 2018.

[24] Security.org Team. Password manager and vault 2021 annual report : Usage, awareness, and market size, 2021.

[25] Verizon. 2022 data breach investigations report, 2022.

[26] Liz Wegerer. Is your browser's password manager safe ?, 2022.

Table des figures

Liste des tableaux

Liste des listings

Annexe A

Journal de travail

TABLE A.1 – Journal de travail

Date	Description	Rech. [h]	Dev. [h]	Rapport [h]	Admin [h]
> 20.09.22	Discussion avec le professeur responsable, établissement du cahier des charges, introduction	7	0	10	4
20.09.2022	Update + organisation du TB, planing, relire le début du TB déjà commencé, avancement de l'étude du marché (fonctionnalités, plateformes, prix), lecture d'articles	2	0	5	1
21.09.2022	Recherches sur les statistiques des gestionnaires de mots de passe sur le marché, rédaction du chapitre étude de marché (terminé ce jour-ci)	3	0	3	0
22.09.2022	Introduction et organisation du chapitre étude sécuritaire, recherche et lecture sur les différentes implémentations sécuritaire des gestionnaires de mots de passe	3	0	1	1
23.09.2022	Recherche et lecture sur les différentes implémentations sécuritaire des gestionnaires de mots de passe et organisation du rapport	1	0	1	0
26.09.2022	Recherche et lecture sur les gestionnaires de mots de passe browser-based, rédaction dans le rapport à ce propos	4	0	1	0
27.09.2022	Recherche et lecture sur les gestionnaires de mots de passe browser-based et local-based, et rédaction dans le rapport	3	0	2	0
28.09.2022	Recherche et lecture sur les gestionnaires de mots de passe browser-based et local-based, et rédaction dans le rapport	4	0	1	0
29.09.2022	Recherche et lecture sur les gestionnaires de mots de passe local-based, et rédaction dans le rapport	5	0	3	0
30.09.2022	Rédaction de la section du partage d'informations, des 3 états du gestionnaires de mots de passe	1	0	4	0
03.10.2022	Fin du chapitre 3 sur analyse de menaces, lecture sur la modélisation de menaces et la norme que je souhaite suivre	7	0	1	0

Le journal de travail continue à la page suivante.

Date	Description	Rech. [h]	Dev. [h]	Rapport [h]	Admin [h]
04.10.2022	Début chapitre analyse de menaces et lecture de la norme choisie	6	0	2	0
05.10.2022	Lecture et rédaction établissement du contexte de l'analyse de menaces	5	0	3	0
06.10.2022	Lecture et rédaction établissement du contexte de l'analyse de menaces	2	0	6	0
07.10.2022	Lecture et rédaction identification des risques de l'analyse de menaces	5	0	3	0
09.10.20220	Lecture et rédaction identification des risques de l'analyse de menaces	2	0	2	0
10.10.20220	Lecture et rédaction identification des risques de l'analyse de menaces	3	0	5	0
11.10.20220	Rédaction et lecture analyses des risques et évaluation des risques	4	0	4	0
12.10.20220	Rédaction et lecture analyses des risques et évaluation des risques	3	0	4	0
13.10.20220	Rédaction du traitement des risques et lecture à ce propos avec les contremesures possibles	3	0	3	0
14.10.20220	Rédaction de la section exigences sécuritaires à respecter, relecture du TB, organisation du rapport et rendu intermédiaire du rapport	1	0	6	0
16.10.20220	Rédaction de la sélection des candidats	2	0	6	0