Hachage & Salage

I/ Qu’est-ce que le hachage

Les fonctions de hachage cryptographique sont un type spécial de calcul unidirectionnel. Ils prennent une chaîne de données de n'importe quelle taille et donnent toujours une sortie d'une longueur prédéterminée. Cette sortie est appelée hacher, valeur de hachage ou résumé du message.
Étant donné que ces fonctions n'utilisent pas de touches, le résultat pour une entrée donnée est toujours le même.

Peu importe si votre entrée est la totalité de *Guerre et Paix* ou simplement deux lettres, le résultat d'une fonction de hachage aura toujours la même longueur. Les fonctions de hachage ont plusieurs propriétés différentes qui les rendent utiles.

I/ Qu’est-ce que le hachage

- **Ce sont des fonctions à sens unique** : Cela signifie qu'il n'existe aucun moyen pratique de déterminer quelle était l'entrée d'origine à partir d'une valeur de hachage donnée.

- **Il est peu probable que deux entrées aient la même valeur de hachage** : Bien qu'il soit possible que deux entrées différentes produisent la même valeur de hachage, les chances que cela se produise sont si faibles que nous ne nous en soucions pas vraiment. À des fins pratiques, les valeurs de hachage peuvent être considérées comme uniques.

- **La même entrée fournit toujours le même résultat -** Chaque fois que vous mettez les mêmes informations dans une fonction de hachage donnée, elle fournira toujours la même sortie.

- **Même le moindre changement donne un résultat complètement différent -** Si même un seul caractère est modifié, la valeur de hachage sera très différente.

II/ Qu’est-ce que le salage

Le **salage** est une méthode permettant de renforcer la sécurité des informations qui sont destinées à être *hachées* (par exemple des mots de passe) en y ajoutant une donnée supplémentaire afin d’empêcher que deux informations identiques conduisent à la même *empreinte* (la résultante d’une fonction de hachage).

Le but du salage est de lutter contre les attaques par analyse fréquentielle, les attaques utilisant des rainbow tables (tables de hachages en libre services ou ayant fuitées), les attaques par dictionnaire et les attaques par force brute. Pour ces deux dernières attaques, le salage est efficace quand le sel utilisé n’est pas connu de l’attaquant ou lorsque l’attaque vise un nombre important de données *hachées* toutes *salées* différemment.

II/ Qu’est-ce que le salage

Le salage consiste à concaténer le mot de passe avec une chaîne de caractères quelconque, le plus souvent aléatoire. Le salage peut être statique : chaque mot de passe est salé avec la même chaîne de caractère (mais ce type de salage est considéré comme dépassé), ou dynamique : chaque mot de passe est salé aléatoirement (cela empêchera à deux utilisateurs d'avoir la même empreinte s'ils ont le même mot de passe).

Dans le cas où le salage est dynamique, chaque enregistrement de la table de mots de passe du système d'authentification contient les informations suivantes :

identifiant | hachage (mot de passe + salage) | salage

III/ Pourquoi utiliser le hachage et le salage de mot de passe

Beaucoup de gens ont de très mauvais mots de passe.
Le problème est que les humains ont tendance à penser selon des modèles prévisibles et à choisir des mots de passe faciles à retenir.

Ces mots de passe sont vulnérables aux attaques de dictionnaire, qui parcourent des milliers ou des millions de combinaisons de mots de passe les plus courantes chaque seconde, dans le but de trouver le mot de passe correct pour un compte.

Si les hachages de mot de passe sont stockés à la place, les choses sont un peu différentes.

Lorsqu'un attaquant rencontre une base de données de hachages de mots de passe, il peut utiliser soit des **tables de hachage** ou **tables arc-en-ciel** pour rechercher des hachages correspondants qu'ils peuvent utiliser pour trouver les mots de passe.

Qu’est-ce qu’une table de hachage ?

Une table de hachage est une liste de hachages pré-calculée pour les mots de passe courants qui est stockée dans une base de données. Ils nécessitent plus de travail à l'avance, mais une fois la table terminée, il est beaucoup plus rapide de rechercher les hachages dans la table que de calculer le hachage pour chaque mot de passe possible. Un autre avantage est que ces tables peuvent être utilisées à plusieurs reprises.

Qu’est-ce qu’une table en arc-en-ciel?

Les tables arc-en-ciel sont similaires aux tables de hachage, elles contiennent toutes les combinaisons possibles des mots de passe en fonction des paramètres choisis à la génération. Ils disposent de correspondances directes mot de passe <-> empreinte.

IV/ De quelle manière s’y prendre

Diverses solutions sont possible afin de mettre en place le hachage et le salage de données tel que :

1) bcrypt

2) Keeweb

La fonction bcrypt

bcrypt est une fonction de hachage créée par Niels Provos et David Mazières.
Elle est basée sur l'algorithme de chiffrement Blowfish de 1991. En plus de l'utilisation d'un salage pour se protéger des attaques par table arc-en-ciel, bcrypt est une fonction adaptative, c'est-à-dire que l'on peut augmenter le nombre de répétitions pour la rendre plus lente. Ainsi elle continue à être résistante aux attaques par brute-force malgré l'augmentation de la puissance de calcul.

Blowfish est un algorithme de chiffrement par bloc, notable pour sa phase d'établissement de clé relativement coûteuse. bcrypt utilise cette propriété et va plus loin. Provos et Mazières ont conçu un nouvel algorithme d'établissement des clefs nommé Eksblowfish (pour *Expensive Key Schedule Blowfish*).
Dans cet algorithme, une première phase consiste à créer les sous-clefs grâce à la clef et au sel. Ensuite un certain nombre de tours de l'algorithme standard blowfish sont appliqués avec alternativement le sel et la clef. Chaque tour commence avec l'état des sous-clefs du tour précédent.

Keeweb

KeeWeb est écrit en JavaScript et utilise WebCrypto et WebAssembly pour traiter les fichiers de mots de passe dans le navigateur, sans les télécharger sur un serveur. Il peut synchroniser les fichiers avec des services d'hébergement de fichiers populaires, tels que Dropbox, Google Drive et OneDrive.

KeeWeb est également disponible sous une forme qui ressemble à une application de bureau. La version de bureau ajoute certaines fonctionnalités qui ne sont pas disponibles sur le web.

logo de Keeweb

Keeweb

WebCrypto :

Web Cryptography est la recommandation du World Wide Web Consortium pour une interface de bas niveau qui augmenterait la sécurité des applications Web en leur permettant d'exécuter des fonctions cryptographiques sans avoir à accéder au matériel de clé brut.

WebAssembly :

WebAssembly, abrégé wasm, est un standard du World Wide Web pour le développement d’applications. Il est conçu pour compléter JavaScript avec des performances supérieures. Le standard consiste en un bytecode, sa représentation textuelle et un environnement d'exécution dans un bac à sable compatible avec JavaScript.

Keeweb

Les différences entre Keeweb et Keepass :

Même principe pour ces deux solutions totalement gratuites : un outil de stockage de mots de passe dans une base de données appelé « vault » (coffre fort) crypté et accessible avec un mot de passe appelé « master password ».

Leur différence majeure étant que KeePass est un logiciel qui s'installe (ce qu'on appelle un « client lourd »), tandis que KeeWeb héberge les mots de passe sur ses serveurs en ligne.

À usage personnel ou professionnel, l'outil s'avère plus que bienvenu lorsqu'il s'agit de devoir partager des mots de passe (sensibles ou pas) avec d'autres personnes.