Université Paul Sabatier
TOULOUSE III

L2 Informatique **Projet S3**

Vérifier mon vote (protocole Debian)

Vincent Dugat - Octobre 2021

Il est impossible dans un vote électronique de vérifier que les votes n'ont pas été falsifiés. Il faudrait renoncer à l'anonymat pour que chacun puisse vérifier que son vote est présent dans la liste officielle du vote et est conforme. La procédure de vote pour élire le chef de projet Debian utilise un procédé qui permet la vérification sans renoncer à l'anonymat.

1 Fonction de hash

Une fonction de hashcoding est une fonction qui associe au moyen d'un calcul une chaîne de caractères de longueur quelconque à une suite de chiffres binaires de longueur constante appelé hashé ou hash. L'opération est facile à calculer mais l'opération inverse [1] est extrêmement complexe voir impossible.

On peut penser qu'il suffit de remplacer le nom des votants par le hashé de leur nom pour garder l'anonymat. Si l'électeur "Astérix" veut vérifier son vote, il calcule le hashé de son nom, le cherche dans la liste des votes et vérifie que son vote est conforme. Le problème est que si Astérix peut le faire avec son nom, n'importe qui peut le faire aussi avec le nom Astérix et savoir ce qu'il a voté.

2 HMAC

L'idée du HMAC est d'ajouter, par concaténation, à la chaîne de caractères que l'on veut asher, une autre chaîne de caractères, générée aléatoirement. Ici il suffit d'ajouter une chaîne aléatoire différente à chaque nom de votant. Cette chaîne aléatoire, si elle n'est connue que de la personne concernée, a une valeur de clef privée.

2.1 Résistance au crack par brute force du codage HMAC

Si on ne peut pas inverser le calcul du hashcoding pour retrouver la chaîne de caractères d'origine, on peut essayer de tester tous les cas possibles. Pour la cas du HMAC, cela signifie concaténer le nom du votant avec les clefs possibles jusqu'à trouver le bon hash. Pour rendre cette opération difficile la clef doit avoir une longueur suffisante pour que le temps de calcul du programme de crack devienne prohibitif.

3 Implémentation

L'implémentation ne va concerner que la vérification du vote. Nous allons utiliser la fonction sha256 qui est utilisée entre autre pour le Bitcoin. Cette fonction n'est pas à programmer, vous allez utiliser une fonction à télécharger sur Moodle.

1. Retrouver la chaîne de caractère connaissant le hash.

Il faut faire une fonction dont les paramètres seront le nom du votant et le fichier de vote. La fonction va demander la clef secrète, concaténer le nom passé en paramètre avec la clef, calculer le hash et cherche ce hash dans le fichier pour trouver la bonne ligne qui sera affichée.

La clef sera donnée sous la forme d'une chaîne de caractères hexadécimaux. Il faudra la convertir en son équivalent en binaire considéré comme des *unsigned char*. Vous pouvez utiliser la fonction *strtoul* avec la base 16.

3.1 Liste des fonctions

— Fonction qui étant donné une chaîne de caractères hexadécimaux, renvoie l'équivalent en binaire (unsigned char en fait).
— Fonction qui étant donné le nom complet de l'électeur et sa clef, concatène les deux dans cet ordre, calcule le hashé et le renvoie.
— Fonction qui étant donné un hash et un fichier de ballots, cherche le hash dans le fichier et renvoie le vote associé. Cette fonction de lecture du fichier csv ressemble beaucoup à celle développée pour les calculs du vote.
— Programme main qui demande (scanf) ou récupère (paramètres de ligne de commande) le nom complet de l'électeur, le nom du fichier de ballots et la clef secrète de l'électeur appelle les fonctions précédentes et affiche le vote de l'électeur.