Université Claude Bernard Lyon 1

LIFPROJET — Printemps 2018 fabrice.mouhartem@ens-lyon.fr

FM3. Système de sondage anonyme

MISE EN GARDE. Ceci est un brouillon incomplet et non relu qui est là pour donner des indications en début de projet. Il sera proprifié par la suite.

1 Introduction

La mise en place de solutions sécurisées de prise de décisions (qui peuvent aller du choix de la date du prochain rendez-vous de travail pour le projet du groupe 43, à celui du dirigeant d'un pays) est une problématique qui a longtemps été résolue par la mise en place de solutions ayant une logistique lourde via un transfert sécurisé des bulletins de vote, et un double dépouillement qui nécessite une certaine quantité de bénévoles pour que cela fonctionne convenablement.

On aimerait donc, dans ces circonstances, une solution plus facile à déployer. Par exemple profitant de la démocratisation de l'ordinateur [1], on pourrait imaginer un site web dont le vote serait sécurisé par l'utilisation de primitives cryptographiques modernes.

C'est pourquoi nous aimerions répondre à ces problématiques de manière *sécurisée* et *flexible*.

2 Boîte à outils cryptographiques

Dans cette section sont décrites des primitives cryptographiques qui pourraient être utiles dans ce projet. C'est-à-dire le chiffrement (partiellement) *homomorphe* (ou *malléable*), les *preuves sans divulgation de connaissance* et le calcul multipartite sécurisé.

2.1 Chiffrement homomorphe

Définition 1. Un schéma de chiffrement asymétrique est la donnée de trois algorithmes fonctionnant en temps polynomial $(\mathsf{GenClefs}, \mathsf{Chiffrer}, \mathsf{Déchiffrer})$, tel que :

- $\mathsf{GenClefs}$ prend en entrée un paramètre de sécurité λ et renvoie une paire de clefs $(\mathrm{pk}, \mathrm{sk})$ dit « clef publique » et « clef privée ».
- $\mathsf{Chiffrer}$ produit un chiffré C à partir d'un message $M \in \{0,1\}^\star$ et d'une clef publique pk.
- $\mathsf{Déchiffrer}$ va renvoyer un message $M \in \{0,1\}^\star$ à partir d'un chiffré C et d'une clef publique sk.

Un chiffrement est *correct* si

$$\Pr_{(pk,sk)\leftarrow\mathsf{GenClefs}(\lambda)}[\mathsf{Déchiffrer}(\mathrm{sk}, \mathsf{Chiffrer}(\mathrm{pk}, M)) = M] = 1 - \lambda^{-\omega(1)}.$$

Définition 2. Un chiffrement est dit *homomorphe* vis-à-vis d'une opération « $\cdot + \cdot$ » s'il existe un algorithme polynomial $\mathsf{Eval}(\cdot,\cdot)$ tel que pour $(\mathrm{pk}, \mathrm{sk}) \leftarrow \mathsf{GenClefs}(\lambda)$, pour tout chiffrés $C_a = \mathsf{Chiffrer}(\mathrm{pk}, a)$ et $C_b = \mathsf{Chiffrer}(\mathrm{pk}, b)$, on a que $\mathsf{Déchiffrer}(\mathrm{sk}, \mathsf{Eval}(C_a, C_b)) = a + b$.

L'homomorphisme est une propriété appréciable dans le cas du vote, par exemple si on souhaite avoir un vote préférentiel « *Oui*/*Non* » et qu'on aimerait avoir le nombre de « *Oui* », alors il suffit d'encoder les « *Oui* » par des 1 et les non par des 0, chiffrer son vote et l'envoyer. Ainsi il ne reste plus qu'à sommer les votes à l'aide d'Eval pour compatibiliser les votes.

Au moment du dépouillement, une autorité déchiffre le vote « Somme », qui agira comme une urne.

1. Le tiers des français posséderait une adresse *e-mail* https://www.arobase.org/actu/chiffres-email.htm

2.2 Preuve à divulgation nulle de connaissance

À venir…
Voir https://paillier.daylightingsociety.org/Paillier_Zero_Knowledge_Proof.pdf

2.3 Calcul multipartite sécurisé

À venir…
Voir https://en.wikipedia.org/wiki/Secure_multi-party_computation

3 Consignes/Pistes

Le but de ce projet sera l'implantation d'un système de *vote*/*sondage* anonyme dont la sécurité et l'anonymat seront garantis par l'utilisation de primitives cryptographiques.

Un point important de ce projet est la mise en place d'une interface utilisateur ergonomique afin que le projet soit utilisable par tous. Par exemple par le biais d'une application web (framadate [2] par exemple), ou d'une application mobile (comme `anonize` [3]). La mise en place d'une API pour permettre le développement d'applications et service tiers peut aussi être envisagée.

Dans un premier temps un court rapport/cahier des charges sera demandé pour validation par le responsable du projet pour évaluer la réalisabilité des solutions proposées.

Ensuite, un déroulement possible serait l'implantation des mécaniques backend via une interface CLI (command line interface) pour permettre les tests unitaires et les tests d'intégration. Cette interface sera ensuite branchée à un *framework* web (par exemple `Flask`) pour permettre un déploiement comme application web.

2. https://framadate.org/
3. https://play.google.com/store/apps/details?id=com.anonize.anonize2