V8TE

Livre scientifique

1. CONTEXTE ET PROBLEMATIQUE

Les élections politiques en Europe sont souvent marquées par un fort taux d'abstention. La plupart des pays de l'UE cherchent aujourd'hui des moyens d'améliorer l'exercice du droit de vote et certains envisagent, à court terme, l'introduction du vote électronique pour juguler l'abstention, automatiser le dépouillement et réduire le coût d'organisation des élections. Toutefois le vote électronique suscite pour l'heure autant d'attentes que de défiance et cette modernisation des processus électoraux ne se fera que si les systèmes de vote proposés présentent de solides garanties de secret et de sécurité et sont ouverts à une vérification indépendante et aisément accessible aux électeurs.

1.1 PROBLEMATIQUE : QUELLE SECURITE POUR LE VOTE EN LIGNE ?

Nous l'observons depuis des siècles, le parcours d'un votant dans un isoloir est extrêmement robuste. Associé à l'enveloppe contenant le bulletin, l'isoloir permet de conserver le secret du vote, de lutter contre la coercition et protège donc la liberté de conscience.

Mettre au point un système de vote électronique sûr est en revanche un exercice beaucoup plus délicat. La difficulté provient de la multiplicité des exigences de sécurité, souvent antinomiques à première vue, qu'un scrutin démocratique doit satisfaire. Par exemple, l'électeur doit être authentifié (pour vérifier qu'il est bien inscrit sur les listes électorales) mais son vote doit rester secret (pour l'éternité[1]). Il doit pouvoir s'assurer non seulement que son bulletin (*vérifiabilité individuelle*), ainsi que ceux des autres votants, est bien présent dans l'urne mais également que le résultat de l'élection correspond bien aux votes exprimés par des électeurs légitimes (*vérifiabilité universelle* ou *collective*). Il doit aussi être incapable de prouver à autrui pour qui il a voté, de manière à décourager toute tentative de corruption ou d'extorsion sous la menace (« *receipt-freeness* » en anglais). Par ailleurs, l'électeur doit avoir la garantie que son vote n'a pas été modifié par une machine à voter (ou un logiciel de vote) malveillant (*vérifiabilité de bout en bout*).

Une question naturelle dès lors se pose : est-il possible de définir un système de vote électronique vérifiant simultanément toutes ces propriétés de sécurité ? La réponse à cette question, qu'elle soit positive ou non, est en effet fondamentale car elle conditionnera les orientations futures prises en matière de vote électronique et son acceptabilité pour certains types de scrutin (vote institutionnel notamment).

Un des résultats-phares de ces dernières années est malheureusement un résultat d'impossibilité. Dans l'article [CFP+10], il est en effet démontré qu'il est impossible, pour un protocole de vote en ligne, de concilier la vérifiabilité du décompte des voix (vérifiabilité universelle) et le secret absolu du vote : un protocole qui garantirait l'une de ces propriétés le ferait nécessairement au détriment de l'autre.

[1] C'est bien souvent par l'utilisation de techniques de chiffrement que l'on arrive à satisfaire le secret du vote dans le contexte du vote électronique. Celles-ci reposent, pour la plupart, sur la difficulté calculatoire de certains problèmes mathématiques issus de la théorie des nombres (on parle dans ce cas de *sécurité calculatoire*). On estime actuellement, par exemple, qu'il est impossible en pratique de factoriser un nombre, tiré aléatoirement, de 3072 bits. Cette hypothèse raisonnable aujourd'hui pourrait très bien être remise en cause dans quelques années en fonction des avancées scientifiques ou technologiques (avènement des ordinateurs quantiques par exemple). Des algorithmes de chiffrement considérés comme sûrs en 2018 pourraient ne plus l'être dans 10 ou 20 ans : il deviendrait dès lors possible de savoir comment chacun des électeurs a voté des années auparavant.

Dans ce même article il a été également démontré formellement que les propriétés de receipt-freeness (« sans reçu » en français) et de vérifiabilité universelle sont incompatibles. Pour les rendre compatibles, il est heureusement possible de faire certaines hypothèses sur le système de vote, comme par exemple, l'existence de *canaux privés ou physiques* entre les votants et l'administrateur de l'élection[2] ou l'absence de collusion entre les différentes entités supervisant l'élection[3].

1.2 LE CONTEXTE NATIONAL : SECRET ABSOLU DU VOTE OU SINCERITE DU SCRUTIN ?

Face à ces limitations, quelles orientations techniques préconisent les autorités en France ou en Europe concernant le vote électronique ?
Le vote *hors-ligne*[4] est autorisé pour des élections politiques depuis le décret n° 69-419 du 10 mai 1969. Plusieurs communes françaises ont d'ailleurs utilisé des machines à voter (différentes de celles retenues pour le marché américain toutefois) lors des présidentielles de 2007 et 2017. L'utilisation de telles machines est aujourd'hui très encadrée. Le ministère de l'Intérieur a en effet élaboré en 2003 une réglementation technique des machines à voter (publiée au journal officiel) qui comporte 114 exigences à respecter.
La situation est beaucoup plus contrastée en France concernant le vote en ligne par Internet. Il est légalement assimilé au vote par correspondance et est de ce fait interdit pour les élections politiques. L'absence d'isoloir, et donc le risque que certains électeurs votent sous la contrainte, constitue la raison principale de cette interdiction. Depuis 2003, il est néanmoins autorisé pour les élections consulaires et prud'homales et tend à se généraliser à bien d'autres types d'élections (vote au sein des assemblées générales d'actionnaires par exemple).
Depuis la Loi pour la Confiance dans l'Economie Numérique du 21 juin 2004, le vote électronique par Internet est également autorisé pour les élections professionnelles. L'utilisation de tels systèmes dans le cadre d'élections professionnelles est toutefois soumise à un certain nombre d'obligations légales (voir par exemple [Ar07, Dec07] pour le détail de ces obligations).
La CNIL (Commission Nationale de l'Informatique et des Libertés), l'une des autorités les plus reconnues en matière de vote électronique, a également émis en 2010, puis plus récemment en avril 2019, un certain nombre de recommandations relatives à la sécurité des systèmes de vote électronique [Del10, Del19]. Ces exigences, notamment celles de 2010, mettent principalement l'accent sur le secret du vote au détriment de la vérifiabilité ou transparence du scrutin. La CNIL précise en particulier dans sa recommandation de 2010 que « *le bulletin de vote doit être chiffré par un algorithme public réputé « fort » dès son émission sur le poste de l'électeur et être stocké dans l'urne, en vue du dépouillement, sans avoir été déchiffré à aucun moment, même de manière transitoire* ». Elle insiste également sur le fait que la solution de vote doit par ailleurs « *garantir que l'identité de l'électeur ne pourra être mise en relation avec l'expression de son vote, et cela à tout moment du processus de vote, y compris après le dépouillement* ». Malheureusement, ces recommandations [Del10] n'abordent pas le problème de la transparence et de la sincérité du scrutin, deux propriétés pourtant essentielles à la démocratie. La CNIL, en effet, ne

[2] Notamment l'envoi, par courrier postal, du *matériel de vote* (éléments d'authentification) aux votants
[3] Hypothèse d'autant plus acceptable que chacune de ces autorités serait représentée par un des candidats en lice.
[4] i.e. l'utilisation de *machines à voter* placées dans des bureaux de vote

prévoit pas expressément que la vérification des résultats soit effectuée par les électeurs eux-mêmes et encore moins qu'ils aient accès au fonctionnement du système mis en place : « *Les fichiers comportant les éléments d'authentification des électeurs, les clés de chiffrement/déchiffrement et le contenu de l'urne ne doivent pas être accessibles, de même que la liste d'émargement, sauf aux fins de contrôle de l'effectivité de l'émargement des électeurs* » [Del 10][5].

1.3 ETAT DE L'ART

Au niveau académique, plusieurs protocoles de vote en ligne ont été élaborés au cours de ces dernières années. Tous ont recours de manière importante à la cryptographie pour garantir à la fois le secret (souvent calculatoire) du vote et/ou la vérifiabilité individuelle (voire universelle) des résultats.

Deux de ces protocoles font aujourd'hui office de référence : le protocole Helios développé sous licence libre par des chercheurs de l'Université d'Harvard et de l'Université catholique de Louvain et le protocole Belenios proposé par des chercheurs de l'INRIA et du CNRS.

Helios a été utilisé à plusieurs reprises lors d'élections étudiantes, par exemple à Princeton et l'Université catholique de Louvain. L'association internationale des chercheurs en cryptographie (IACR) l'a également retenu pour élire les membres de son bureau. Le système Helios assure la vérifiabilité universelle des résultats et le secret calculatoire du vote. Il ne respecte malheureusement pas les recommandations de 2010 de la CNIL et pour cette raison ne recevrait sans doute pas d'agrément en France pour être utilisé lors d'élections officielles. En effet, afin de satisfaire la propriété de vérifiabilité universelle, Helios rend publique *la liste d'émargement* et permet, de manière indirecte, la *mise en relation de l'identité de l'électeur avec l'expression de son vote*[6].

Belenios est une évolution d'Helios qui offre une meilleure protection que ce dernier contre le bourrage d'urne (y compris face à un attaquant qui contrôlerait une des autorités en charge de l'élection). Belenios assure également la vérifiabilité universelle des résultats et le secret calculatoire du vote mais tout comme Helios, et pour des raisons analogues, ne respecte pas les préconisations de 2010 de la CNIL.

Il existe également des protocoles de vote en ligne garantissant le secret absolu du vote [CPP13, DGS12] mais ces derniers ne respectent pas non plus les préconisations de la CNIL : il est en effet nécessaire, avec ce type de protocoles, de publier la liste d'émargement pour garantir une certaine transparence du scrutin.

De nombreuses solutions propriétaires ont également vu le jour [Doc07, Vox06] depuis 2003, date de l'autorisation du vote en ligne pour les élections consulaires et prud'homales. Si pour la plupart, ces solutions respectent le cahier des charges de la CNIL, elles n'apportent pas la transparence qu'exige tout scrutin démocratique : les électeurs n'ont ainsi pas la possibilité de surveiller en permanence l'urne (virtuelle) jusqu'au dépouillement, comme pour une élection traditionnelle, et n'ont donc pas l'assurance que leur bulletin y figure bien, que personne ne l'a modifié et que les personnes ayant voté avaient bien le droit de le faire.

[5] Il est à noter toutefois que la recommandation récente de 2019 intègre maintenant les notions de transparence du scrutin et de vérifiabilité collective : voir par exemple dans [Del19], l'« *Objectif de sécurité n° 2-07 : Assurer la transparence de l'urne pour tous les électeurs* » ainsi que l'« *Objectif de sécurité n° 3-02 : Permettre la transparence de l'urne pour tous les électeurs à partir d'outils tiers* ».

[6] Ce qui fait courir le risque que l'on puisse savoir pour qui un électeur a voté, dans le cas où les mécanismes cryptographiques utilisés par Helios ne seraient plus sûrs.

2. NOTRE INVENTION

Nous proposons un nouveau protocole de vote qui offre la possibilité à n'importe quel votant de s'assurer que son bulletin n'a pas été falsifié (i.e. que son contenu et donc son choix n'a pas été modifié) et qu'il a été correctement comptabilisé (vérifiabilité individuelle). Tout électeur peut également s'assurer que les autres bulletins proviennent d'électeurs légitimes et vérifier la validité des résultats proclamés par les membres du bureau de vote (vérifiabilité universelle). Notre protocole de vote garantit par ailleurs à tout électeur le secret absolu de son choix[7]. Cette garantie est accentuée, lorsqu'après le délai de recours légal, les bulletins sont définitivement détruits.
Notre solution respecte en outre les recommandations de la CNIL[8] et va même au-delà en offrant un maximum de transparence aux votants et le secret absolu de leur vote.

2.1 LES OUTILS CRYPTOGRAPHIQUES

Pour offrir de telles garanties de sécurité, nous nous appuyons sur un ensemble de techniques cryptographiques largement éprouvées : un schéma de chiffrement *homomorphe additif à seuil*, un schéma de mise en gage (*commitment scheme* en anglais) homomorphe additif, des preuves à divulgation nulle de connaissance (Zero Knowledge Proof of Knowledge, notées ZKPK dans la suite), des réseaux de mélangeurs (mix-networks en anglais) ainsi que des signatures anonymes. Nous présentons très brièvement ces différents concepts dans les sections suivantes (en essayant d'occulter le plus possible le formalisme mathématique sous-jacent).

2.1.1 Chiffrement homomorphe additif à seuil

Pour garantir le secret du vote et respecter les recommandations de la CNIL[9], nous utiliserons un algorithme de chiffrement à clé publique *homomorphe* pour l'*addition*. On rappelle qu'un tel mécanisme permet à partir d'un chiffré d'un message m_1 et d'un chiffré d'un message m_2 d'obtenir un chiffré du message $m_1 + m_2$. Nous choisirons en outre un *algorithme de chiffrement* à *seuil*[10]. Pour ce type particulier d'algorithmes de chiffrement, la clé privée s_k associée à une clé publique P_k est un secret réparti entre différents participants. Pour déchiffrer n'importe quel message, la coopération d'un nombre prédéfini (*le seuil*) de ces participants est requise. En notant s le seuil et n le nombre de participants entre lesquels le secret est réparti, on ne peut déchiffrer un message que si au moins s participants sur les n coopèrent, et ce sans qu'il soit nécessaire de reconstituer la clé privée. Le schéma de chiffrement de Paillier [Pai99], introduit à la fin des années 90, est un exemple de schéma de *chiffrement homomorphe additif à seuil*. Ainsi, en multipliant un chiffré de Paillier d'un

[7] Sauf pour les administrateurs de l'élection pour lesquels cette sécurité ne sera que calculatoire : si ces derniers réussissent à casser le schéma de chiffrement utilisé, ou s'ils sont tous de connivence, alors ils sauront pour qui chaque électeur a voté
[8] Nous ne dévoilons ni le contenu de l'urne ni la liste d'émargement et « garantissons que l'identité de l'électeur ne pourra être mise en relation avec l'expression de son vote, et cela à tout moment du processus de vote, y compris après le dépouillement »
[9] « le bulletin de vote doit être chiffré par un algorithme public réputé « fort » dès son émission sur le poste de l'électeur et être stocké dans l'urne, en vue du dépouillement, sans avoir été déchiffré à aucun moment, même de manière transitoire » [Del10].
[10] Cet algorithme « *permettra le dépouillement exclusivement au sein du bureau électoral et garantira la possibilité de dépouillement à partir d'un seuil de secret déterminé »* (voir [Del19], objectif de sécurité n°1-08).

message m_1 par un chiffré de Paillier d'un message m_2, on obtient un chiffré du message m_1+ m_2 : $E_{P_k}(m_1) \times E_{P_k}(m_2) = E_{P_k}(m_1 + m_2)$ où E désigne l'algorithme de chiffrement de Paillier et P_k une clé publique.

2.1.2 Les schémas de mise en gage

La mise en gage d'une valeur est un procédé cryptographique permettant à un émetteur (Alice) de s'engager sur une valeur (un secret par exemple) auprès d'un destinataire (Bob) sans la dévoiler de prime abord et de telle sorte que cet engagement ne soit plus modifiable *a posteriori*. Cette valeur peut être, si besoin est, dévoilée par la suite par l'émetteur.
Ainsi, le destinataire a l'assurance qu'une fois l'engagement publié, l'émetteur ne peut plus changer d'avis sur la valeur contenue dans cet engagement. De manière imagée, une mise en gage consiste à remettre à Bob un coffre fermé renfermant le secret, et à lui fournir ultérieurement la combinaison permettant de l'ouvrir.
Exemple : une façon simple de construire un schéma de mise en gage consiste à utiliser une fonction de hachage H. On rappelle qu'une telle fonction doit être *résistante aux collisions* : en d'autres termes, il doit être difficile en pratique de trouver deux messages m et m' tels que $H(m) = H(m')$. Une telle fonction doit être également difficile à *inverser* : un condensé C étant donné, il doit être difficile en pratique de trouver un message m tel que $C = H(m)$.
Pour s'engager auprès de Bob sur une valeur *a priori* secrète x (la combinaison gagnante du prochain tiercé par exemple), Alice choisit un aléa r et calcule l'engagement $C = H(x||r)$ qu'elle envoie à Bob.[11] Ce faisant, Alice ne peut plus changer d'avis. Dans le cas contraire, cela voudrait dire qu'elle a pu trouver une autre valeur $x' \neq x$ (ainsi qu'un autre aléa r') telle que $C = H(x'||r')$, ce qui contredirait l'hypothèse selon laquelle la fonction H est résistante aux collisions.
Par ailleurs, Bob n'est pas en mesure, à partir de C, de retrouver la valeur qui a été mise en gage. Dans le cas contraire, cela signifierait qu'il est en mesure d'*inverser* la fonction de hachage H.[12]
A l'issue de la course, Alice peut prouver à Bob qu'elle avait bien le tiercé dans l'ordre en lui révélant la paire (x, r). Bob peut s'assurer de sa bonne foi en vérifiant que $C = H(x||r)$.
Nous utiliserons pour notre protocole de vote, le schéma de mise en gage proposé par Pedersen en 1992 [Ped92]. Ce schéma a la particularité de produire des engagements *parfaitement non distinguables* (perfectly hiding en anglais)[13] et *calculatoirement résistants aux collisions*[14] et d'être homomorphe pour l'addition. Avec ce schéma, en multipliant un engagement C_1 sur une valeur s_1 par un engagement C_2 sur une valeur s_2, on obtient un nouvel engagement C sur une valeur s_1+s_2.

[11] Le symbole || désigne le symbole de concaténation.
[12] L'utilisation de l'aléa *r* est très importante. Étant donné qu'il n'y a qu'un nombre restreint de combinaisons possibles, Bob aurait pu retrouver par recherche exhaustive la combinaison gagnante x, si Alice s'était contentée de lui envoyer $C = H(x)$ au lieu de $C = H(x, r)$.
[13] Un attaquant en possession d'une mise en gage *C* n'apprendra aucune information sur la valeur secrète (le choix d'un votant par exemple) qui a été mise en gage, même s'il dispose d'une puissance de calcul non bornée. C'est sur cette propriété que nous nous appuyons pour démontrer que notre protocole de vote garanti le *secret absolu du vote*.
[14] un engagement C d'un message *m* étant donné, il difficile en pratique de trouver un autre message m' tel que C soit également un engagement sur le message m'.

2.1.3 Preuves à divulgation nulle de connaissance

Les électeurs devront au moment de voter, prouver, en utilisant des preuves de connaissance dites à divulgation nulle de connaissance (ZKPK en anglais) [GMR85], que leur vote (qui sera chiffré) porte bien sur un des candidats à l'élection. En outre, les assesseurs devront prouver qu'ils ont correctement déchiffré le résultat de l'élection sans dévoiler leurs clés privées de déchiffrement.

Une ZKPK est un protocole entre un « prouveur » et un « vérifieur » permettant au vérifieur de se convaincre que le prouveur connaît un secret *S* satisfaisant un prédicat *P* donné. En outre, la preuve ne révèle au vérifieur aucune information sur le secret *S* en question hormis le fait que celui-ci vérifie bien le prédicat *P* donné.

Exemple : une illustration de la manière dont se déroule une preuve à divulgation nulle de connaissance a été donnée par Gradwohl et al. [GNPR09, section 4.1]. Elle s'applique au jeu du Sudoku.

Alice veut convaincre Bob qu'elle a réussi à compléter la grille de Sudoku suivante (Figure 1) mais ne souhaite pas lui dévoiler la solution (Figure 2) qu'elle a obtenue.

1								6
		6		2		7		
7	8	9	4	5		1		3
			8		7			4
				3				
	9				4	2		1
3	1	2	9	7			4	
	4			1	2		7	8
9		8						

Figure 1

1	2	3	7	8	9	4	5	6
4	5	6	1	2	3	7	8	9
7	8	9	4	5	6	1	2	3
2	3	1	8	9	7	5	6	4
5	6	4	2	3	1	8	9	7
8	9	7	5	6	4	2	3	1
3	1	2	9	7	8	6	4	5
6	4	5	3	1	2	9	7	8
9	7	8	6	4	5	3	1	2

Figure 2

Dans un premier temps Alice va reproduire à l'aide de cartes à jouer (seules celles numérotées de 1 à 9 seront toutefois utilisées) la grille de la Figure 2 (en mettant face cachée, les cartes correspondant à la solution qu'elle a trouvée). Une fois la grille reconstituée (Figure 3), Bob est invité à choisir au hasard une ligne, une colonne ou un carré (par exemple la sixième ligne).

1								6
		6		2		7		
7	8	9	4	5		1		3
			8		7			4
				3				
	9				4	2		1
3	1	2	9	7			4	
	4			1	2		7	8
9		8						

Figure 3

Alice récupère les cartes (dont la face est cachée) de la ligne choisie par Bob, puis les mélange avant de les remettre à Bob.
Bob doit s'assurer que la sixième ligne (constituée des cartes déjà retournées et celles dévoilées par Alice) contient bien tous les chiffres de 1 à 9 et que ceux-ci n'apparaissent bien qu'une seule fois (Figure 4). Après plusieurs tests concluants (en choisissant au hasard de nouvelles lignes ou de nouvelles colonnes), Bob aura la ferme conviction qu'Alice connaît bel et bien une solution à cette grille de Sudoku. Toutefois les réponses fournies par Alice ne lui seront d'aucune aide pour résoudre à son tour ce casse-tête mathématique. En effet, le fait qu'Alice ait brassé les cartes (dont la face était cachée) avant de les transmettre à Bob empêche ce dernier de savoir comment elles étaient initialement disposées sur la grille.

1								6
		6		2		7		
7	8	9	4	5		1		3
			8		7			4
				3				
	9				4	2		1
3	1	2	9	7			4	
	4			1	2		7	8
9		8						

5	3	8	6	7

Figure 4

2.1.4 Réseau de mélangeurs

Le concept de réseaux de mélangeurs (ou mix-networks en anglais) a été inventé par David Chaum [Cha81] dans les années 1980 pour garantir la confidentialité des votes dans les systèmes de vote électronique. Un mélangeur est un serveur qui prend en entrée un nombre arbitraire de données et qui a pour but de cacher la correspondance entre ces données et celles qu'il produira en sortie. L'utilisation de plusieurs mélangeurs en série (on parle dans ce cas de *réseaux de mélangeurs*) permet d'être sûr du résultat final dès lors qu'un des mélangeurs a réellement brassé les données.
En pratique les données reçues en entrée sont des données chiffrées (les votes des électeurs) et le brassage effectué par le réseau permet de ne plus pouvoir tracer qui a chiffré quoi. On exige généralement des mélangeurs qu'ils prouvent qu'ils ont bien fait leur travail, c'est-à-dire que le brassage a bien été effectué et qu'aucune donnée chiffrée n'a été

modifiée, rajoutée ou supprimée. La technique proposée par Jakobsson, Juels, and Rivest, connue sous le nom de RPC (Random Partial Checking) [JJR02], est l'une des plus efficaces à l'heure actuelle pour réaliser ce type de preuve à divulgation nulle de connaissance. Dans le contexte de notre protocole de vote, cette technique sera utilisée pour anonymiser les clés publiques d'émargement des votants.

2.1.5 Signatures anonymes

Les votants devront prouver qu'ils sont bien autorisés à prendre part à un scrutin sans toutefois dévoiler leur identité. Nous utiliserons à cet effet un schéma de *signatures anonymes* (une variante d'un schéma de *signature de groupe*). De tels schémas permettent à un utilisateur de prouver son appartenance à un groupe (par exemple d'électeurs, de personnes ayant souscrit à un service, etc.) sans qu'il ait à révéler son identité exacte. Les signatures de groupe ont la particularité d'être *anonymes* (on ne peut identifier le signataire) et *non traçables* (on ne peut pas déterminer si deux signatures ont été émises par la même personne ou par deux personnes distinctes). La validité d'une signature de groupe peut être vérifiée par quiconque grâce à une clé publique caractérisant le groupe (appelée "clé publique de groupe"). Pour faire partie du groupe, un membre doit s'enregistrer auprès d'un *manager de groupe*. Lors de cette phase d'enregistrement, le futur membre obtient, de *manière aveugle*[15], une *clé privée* (*de groupe*) qui va lui permettre de signer des messages au nom du groupe. Seule une autorité de confiance a le pouvoir de révoquer l'anonymat d'une signature de groupe, grâce à une *trappe* (une clé privée particulière) qu'elle seule possède. En pratique, cette trappe est partagée entre plusieurs autorités de confiance (autorité de révocation) et il est nécessaire qu'elles coopèrent toutes pour lever l'anonymat d'une signature. Le membre du groupe se voit ainsi protéger contre des levées d'anonymat abusives.
Dans le contexte du vote, ces autorités n'accepteraient, par exemple, de lever l'anonymat d'un votant qu'à la demande explicite d'un juge.
Dans le contexte de notre protocole de vote, nous utiliserons en fait une variante des signatures de groupe qui, à la différence de ces dernières, sont anonymes mais *traçables* [BD+17]. Dans le contexte du vote, cette traçabilité nous permettra de nous assurer qu'un électeur ne votera qu'une seule fois.

2.2 PRESENTATION DES PARTIES PRENANTES

Nous reprenons en partie le modèle d'acteurs décrit dans le « Référentiel définissant les attentes relatives aux solutions de vote par Internet » rédigé par la Fédération des Tiers de Confiance (FNTC).

Plusieurs entités sont indispensables pour notre système de vote :

- Les électeurs
- Les candidats
- Le bureau de vote
- L'organisateur
- L'autorité d'enregistrement
- Une urne privée
- Une urne publique

[15] Le manager de groupe ne connaîtra donc pas la clé privée de signature obtenue par l'utilisateur.

2.2.1 Les électeurs

Ils expriment leurs choix au travers de leurs suffrages et émargent (de manière anonyme) au moment de voter. Ils peuvent s'assurer que leur bulletin sera bien pris en compte en vérifiant qu'un « engagement » sur leur vote apparaît bien sur l'urne publique. Ils ont également la possibilité d'auditer a posteriori le dépouillement grâce aux données publiées sur l'urne publique.

2.2.2 Les candidats

Ils présentent leur candidature et surveillent le bon déroulement du scrutin

2.2.3 Le bureau de vote

Il est constitué d'au moins 3 assesseurs indépendants [Del10].
Eux seuls peuvent procéder au dépouillement du scrutin. Chaque membre du bureau dispose d'une clé de déchiffrement et il faut en réunir au moins deux sur les 3 pour procéder au dépouillement du scrutin [Del10].

2.2.4 L'organisateur

Il fournit les données de référence (liste électorale) et reçoit les résultats

2.2.5 L'autorité d'enregistrement

Elle a en charge la fourniture du matériel de vote et l'envoi de ce matériel aux électeurs.
Elle est également chargée, en collaboration avec l'organisateur, d*'anonymise*r la liste électorale et/ou de fournir les *clés privées de groupe* aux électeurs dûment enregistrés sur la liste électorale.
Elle peut produire, en collaboration avec l'organisateur, et à la demande d'un juge, la liste d'émargement. Ce rôle d'autorité d'enregistrement peut être exercé, pour plus de sécurité, par plusieurs entités (et non une seule).

2.2.6 L'urne privée

Il s'agit d'un serveur administré par le bureau de vote et uniquement accessible par les membres de ce bureau comme l'exige la CNIL [Del10]. Son rôle est de collecter les bulletins émis par les électeurs, de les vérifier et de ne conserver que les bulletins valides. Chaque bulletin valide de notre protocole de vote sera constitué de 5 éléments, C_1, C_2, C_3, P et S :

- C_1 est un engagement sur le choix v du votant ainsi que sur une valeur aléatoire r choisie par ce dernier ;
- C_2 un chiffré de v ;
- C_3 un chiffré de r ;
- P une ZKPK prouvant notamment que la valeur v mise en gage dans C_1 correspond bien à un vote pour un des candidats en lice ;
- S une signature (anonyme) de l'électeur sur C_1 et P.

L'urne privée doit transmettre, pour chaque bulletin valide, la mise en gage C_1, la preuve P et la signature S correspondante, à l'urne publique.
L'administration de cette urne privée peut se faire de manière décentralisée selon le principe d'une blockchain privée.

2.2.7 L'urne publique

Son rôle est de conserver les « éléments de preuve » (les engagements et les signatures transmises par l'urne privée) qui permettront d'auditer l'élection.
Afin d'offrir plus de transparence aux électeurs et de garantir la sincérité du scrutin et la vérifiabilité collective, les engagements et les signatures anonymes sont enregistrées sur un DLT qui fait office d'urne publique. L'empreinte (« Hash ») certifiant l'authenticité de cette urne publique est enregistrée sur une blockchain publique.
Avec ce DLT, tous les citoyens pourront accéder à ces « éléments de preuve » et ainsi vérifier le bon déroulement et l'intégrité des scrutins et des résultats [Del19, objectifs de sécurité n°2-07 et 3-02].

2.3 LES PRINCIPALES PHASES

Nous considèrerons 5 phases principales lors d'une opération de vote :

- La préparation du scrutin
- Le scrutin
- Le dépouillement
- L'audit
- L'archivage et les recours

2.3.1 La préparation du scrutin

C'est au cours de cette phase que l'organisateur constitue la liste électorale ainsi que la liste des candidats. Les paramètres publics ainsi que les clés des différents participants (électeurs, organisateur, autorité d'enregistrement, membres du bureau de vote) sont générés également lors de cette phase.

2.3.2 Le scrutin

C'est uniquement durant cette phase que les électeurs expriment leurs choix. L'électeur doit transmettre son bulletin et son émargement à l'urne privée. Il pourra s'assurer que son bulletin a bien été reçu en vérifiant que l'engagement (C_1) sur son vote (v) est bien présent dans l'urne publique.

2.3.3 Le dépouillement

A la clôture du scrutin, les membres du bureau de vote procèdent au décompte des votes et proclament les résultats. Ils fournissent également les éléments de preuves permettant d'auditer l'élection et d'attester de la validité des résultats proclamés.

2.3.4 L'audit

Toute personne (électeur, candidat, observateur indépendant, etc.) a la possibilité d'auditer l'élection grâce aux informations publiées sur l'urne publique et aux éléments de preuve fournis par les membres du bureau de vote.

2.3.5 L'archivage et les recours

Les membres du bureau de vote doivent archiver des éléments de traçabilité (notamment tous les bulletins collectés par l'urne privée) pendant le délai de recours. En cas de recours, ces éléments permettront notamment un recomptage des votes émis. Ils doivent dans tous les cas être détruits à l'expiration des délais de recours.

Le recomptage pourra être fait à partir de l'urne publique transparente après la destruction règlementaire de l'urne privée opaque. Pour autant, la liste d'émargement devra être auditée avant la fin du délai de recours sur décision d'un juge.

2.4 PRINCIPE DE FONCTIONNEMENT DU PROTOCOLE DE VOTE

Nous considérerons dans la suite un scrutin particulier : le référendum (où les deux choix possibles seront « 1 », pour le « oui », et « 0 », pour le « non »). Les techniques classiques permettant de prendre en compte des scrutins plus complexes (tels que le jugement majoritaire, le scrutin uninominal majoritaire, etc.) s'appliquent également à notre système de vote.

Pour un vote avec plusieurs candidats, chaque candidat est considéré comme un référendum sur son nom[16]. Le protocole du vote sera la combinaison des référendums sur chaque nom en s'assurant qu'un seul nom est choisi pour chaque vote.
Nous supposerons qu'il y a n électeurs inscrits sur les registres électoraux et que le bureau de vote est constitué de k membres.

2.4.1 La préparation du scrutin

Dans un premier temps, les membres du bureau de vote ainsi que l'organisateur et l'autorité d'enregistrement doivent s'accorder sur les algorithmes à clé publique qui seront utilisés lors de ce scrutin : un algorithme de *chiffrement homomorphe additif à seuil* (E), un schéma de *mise en gage homomorphe pour l'addition* $(Commit)$ et un schéma de *signatures anonymes* $(SigA)$.

Chaque membre (M_i) du bureau de vote génère sa clé privée (SK_{M_i}) pour l'algorithme de chiffrement homomorphe additif à seuil E.

En parallèle, l'Organisateur (O) et l'Autorité d'enregistrement (A), qui feront office de managers de groupe pour l'algorithme de signatures anonymes $SigA$, génèrent leurs paires de clés respectives, (SK_O, PK_O) et (SK_A, PK_A), pour cet algorithme.

Lors de cette phase de préparation du scrutin, chaque électeur V_i dûment inscrit sur la liste électorale, obtient, de *manière aveugle*, de l'Organisateur et de l'Autorité d'enregistrement, une clé privée SK_{V_i} de l'algorithme $SigA$.

Les paramètres et clés publiques de ces différents algorithmes sont publiés et serviront lors de la phase de vote proprement dite ainsi que pour la phase d'audit de l'élection.

2.4.2 Le scrutin

Le votant V_i , où i est compris entre 1 et n, va dans un premier temps générer une valeur aléatoire r_i, puis calculer un engagement sur son choix v_i (i.e. «0 » ou « 1 ») ainsi que sur la valeur aléatoire r_i : $C_1^i = Commit(v_i, r_i)$

V_i va ensuite chiffrer son choix v : $C_2^i = E_{PK}(v_i)$ où E désigne l'algorithme de chiffrement homomorphe additif à seuil retenu pour ce scrutin et PK la clé publique associée. Comme il s'agit d'un schéma de chiffrement à seuil, la clé privée SK associée à PK est partagée entre les k membres du bureau de vote ; chaque membre en possède un morceau (sa clé SK_{M_i}).

[16] Le votant sera donc invité à voter « oui », c'est-à-dire « 1 », s'il choisit ce candidat et « non », c'est-à-dire « 0 » dans le cas contraire. Le votant devra donc se prononcer *k* fois s'il y a *t* candidats et devra prouver (en utilisant une ZKPK) qu'il n'a voté qu'une seule fois pour le « oui » lors de ces *t* référendums.

La coopération de s d'entre eux est requise pour déchiffrer n'importe quel message. En particulier, il sera nécessaire que s d'entre eux coopèrent pour déchiffrer le résultat de l'élection.

V_i va ensuite chiffrer la valeur aléatoire r_i : $C_3^i = E_{PK}(r_i)$ [17]

Puis il va générer deux preuves (ZKP_1, ZKP_2) à divulgation nulle de connaissance[18] prouvant que

1. $(\boldsymbol{ZKP_1})$ le texte clair v_i associé au chiffré $C_2^i = E_{PK}(v_i)$ vaut bien « 0 » ou « 1 ». Ceci afin d'éviter qu'un votant indélicat calcule volontairement un chiffré de la valeur « 2 » par exemple ; ce qui équivaudrait à voter deux fois pour le « oui » en un seul vote;
2. $(\boldsymbol{ZKP_2})$ les textes clairs associés aux chiffrés C_2^i et C_3^i (à savoir v_i et r_i) sont bien les mêmes que ceux qui ont été mis en gage dans l'engagement C_1^i.

V_i va ensuite signer (anonymement) le chiffré C_1^i ainsi que ZKP_1 grâce à sa clé privée SK_{V_i} de l'algorithme $SigA$. Cette signature S_i correspond à l'émargement qu'un électeur doit apposer sur la liste d'émargement dans le cadre d'une élection traditionnelle. La différence notable est que dans notre cas cette signature est anonyme et ne permettra donc pas d'identifier l'électeur qui vient de voter.

Le votant V_i n'a plus qu'à envoyer son bulletin B_i constitué, de l'engagement C_1^i, des chiffrés C_2^i et C_3^i, des deux preuves (ZKP_1, ZKP_2) ainsi que de la signature S_i, à l'urne privée.

Pour chaque nouveau bulletin reçu, l'urne doit vérifier la validité des deux preuves ainsi que celle de la signature S_i. Elle doit également s'assurer que le bulletin provient d'un électeur légitime qui n'a pas déjà voté.

Cette vérification est aisée. En effet, la signature S_i prouve qu'elle a bien été émise par un électeur inscrit sur la liste électorale. Les signatures anonymes sont par ailleurs traçables : si l'électeur a déjà voté, l'urne pourra retrouver parmi les bulletins qu'elle a déjà reçus, une autre signature (S_i') émise par ce même électeur. L'urne ne retiendra cependant qu'un seul des deux bulletins : le tout premier (celui contenant la signature S_i').

Le bulletin est considéré comme valide si toutes ces vérifications sont concluantes. L'urne privée doit ensuite transmettre, pour chaque bulletin valide B_i, la mise en gage C_1^i, la preuve ZKP_1 et la signature S_i correspondante, à l'urne publique.

2.4.3 Le dépouillement

A la clôture du scrutin, les membres du bureau de vote vont récupérer tous les bulletins valides collectés par l'urne privée.

Chaque membre va ensuite :

1. multiplier[19] entre eux tous les engagements C_1^i pour obtenir, grâce au caractère homomorphe du schéma de mise en gage $Commit$, un nouvel engagement R_1

[17] L'algorithme de chiffrement utilisé pour chiffrer r_i pourrait être différent de celui utilisé pour chiffrer v. Par souci de simplification, nous supposerons que c'est le même algorithme qui sera utilisé pour chiffrer ces deux valeurs.

[18] En pratique, on utilise des « *signatures de connaissances* » qui vont en plus permettre de s'assurer que les preuves ZKP_1 et ZKP_2 portent bien sur le triplet (C_1^i, C_2^i, C_3^i) qui est composé d'éléments indissociables du bulletin de vote de V_i.

[19] Nous supposerons par souci de simplification que c'est en multipliant un engagement C_1 sur une valeur s_1 par un engagement C_2 sur une valeur s_2, que l'on obtient un nouvel engagement C de la valeur s_1+s_2.

portant sur la somme des choix v_i effectués par les électeurs ayant pris part au scrutin (mais également sur la somme des aléas r_i tirés aléatoirement par ces électeurs) : $R_1 = \prod_{i=1}^{l} C_1^i = \prod_{i=1}^{l} Commit(v_i, r_i) = Commit\left(\sum_{i=1}^{l}(v_i), \sum_{i=1}^{l}(r_i)\right)$ où l désigne le nombre de bulletins valides reçus par l'urne privée.

2. multiplier[20] entre eux tous les bulletins chiffrés C_2^i pour obtenir, grâce au caractère homomorphe du schéma de chiffrement E, un nouveau chiffré R_2 correspondant à la somme des choix v_i effectués par les électeurs ayant pris part au scrutin : $R_2 = \prod_{i=1}^{l} C_2^i = \prod_{i=1}^{l} E_{PK}(v_i) = E_{PK}\left(\sum_{i=1}^{l}(v_i)\right)$.
3. multiplier entre eux tous les bulletins chiffrés C_3^i pour obtenir, grâce au caractère homomorphe du schéma de chiffrement E, un nouveau chiffré R_3 correspondant à la somme des aléas r_i tirés aléatoirement par les électeurs ayant pris part au scrutin : $R_3 = \prod_{i=1}^{l} C_3^i = \prod_{i=1}^{l} E_{PK}(r_i) = E_{PK}\left(\sum_{i=1}^{l}(r_i)\right)$.

Chaque membre M_i du bureau de vote va ensuite procéder au déchiffrement partiel de R_2 (respectivement R_3) à l'aide de sa clé privé SK_{M_i}. Dès qu'on a obtenu s (le seuil) déchiffrements partiels du message R_2, il est possible, en les combinant, de retrouver le texte clair $Res = \sum_{i=1}^{l}(v_i)$ (respectivement $Rand = \sum_{i=1}^{l}(r_i)$) qui correspond au nombre de voix qui se sont portées pour le « oui ». On obtient donc le résultat de ce référendum.
Les k membres (M_i) du bureau de vote publient Res et $Rand$.

N.B. Les engagements C_1^i publiés sur l'urne publique ne révèlent aucune information sur les choix effectués par les électeurs puisque le schéma de mise en gage $Commit$ produit des engagements *parfaitement non distinguables*. Le secret du vote est donc parfait pour toute personne extérieure au bureau de vote[21]. Par ailleurs il n'est pas possible de procéder au dépouillement des bulletins de vote avant la fin du scrutin si au moins $k - s + 1$ membres du bureau de vote respectent la règle, puisqu'il faut la présence d'au moins s d'entre eux pour dépouiller un bulletin de vote. En particulier ces $k - s + 1$ membres refuseront de déchiffrer des bulletins individuels et n'accepteront de déchiffrer que les chiffrés R_2 et R_3 qui contiennent le résultat de l'élection.

2.4.4 L'audit

Le résultat de l'élection est vérifiable par tous. Pour s'assurer de la validité du résultat Res proclamé, un observateur doit après avoir récupéré la valeur $Rand$:

1. vérifier que tous les bulletins tronqués[22] publiés sur l'urne publique proviennent bien d'électeurs légitimes et que ceux-ci n'ont voté qu'une seule fois. Cette vérification est faite grâce aux signatures S_i qui accompagnent les engagements C_1^i. Chaque signature S_i doit être valide et on ne doit pas trouver dans l'urne publique plusieurs signatures émises par un même électeur.
2. multiplier entre eux tous les engagements C_1^i pour obtenir, grâce au caractère homomorphe du schéma de mise en gage $Commit$, un nouvel engagement R'_1 : $R'_1 = \prod_{i=1}^{\bar{l}} C_1^i =$ où $\bar{l}$ désigne le nombre de bulletins tronqués reçus par l'urne publique.

Si aucune anomalie n'a été détectée à l'étape 1 et que $R'_1 = Commit(Res, Rand)$ alors c'est que le dépouillement a été effectué de manière correcte.

[20] Nous supposerons également par souci de simplification que c'est en multipliant un chiffré d'un message m_1 par un chiffré d'un message m_2 que l'on obtient un chiffré du message $m_1 + m_2$.
[21] Un attaquant même en disposant d'une puissance de calcul non bornée ne pourra pas à partir de l'engagement *Commit* savoir sur qui s'est porté le choix de l'électeur.
[22] constitué d'un engagement C_1^i, de la preuve ZKP_1 et d'une signature S_i sur C_1^i et ZKP_1.

2.4.5 L'archivage et les recours

Les membres du bureau de vote doivent archiver des éléments de traçabilité (notamment tous les bulletins collectés par l'urne privée). En cas de recours, ces éléments doivent permettre un recomptage des votes émis. Ils doivent être détruits à l'expiration des délais de recours.

Variante : dans cette variante, l'urne privée, au lieu de stocker dans leur intégralité tous les bulletins collectés, va les « agréger » au fur et à mesure. Elle ne conservera donc que des bulletins « agrégés » et en aucun cas des bulletins individuels : ceux-ci seront « détruits », au fil de l'eau, juste après avoir été transmis à l'urne publique (sous réserve qu'ils soient valides). Il ne sera ainsi **plus possible** y compris pour les membres de vote, en cas de collusion de leur part, de relier *l'identité d'un électeur à l'expression de son vote* [Del10] et donc de déterminer pour qui ce dernier avait voté.

Plus précisément, soit l le nombre de bulletins déjà collectés par l'urne privée. Désignons par $B_i = (C_1^i, C_2^i, C_3^i, C_1^i, ZKP_1^i, ZKP_2^i, S_i)$ l'un de ces bulletins et par $R_2^l = \prod_{i=1}^{l} C_2^i = \prod_{i=1}^{l} E_{PK}\ (v_i) = E_{PK}\left(\sum_{i=1}^{l}(v_i)\right)$ (respectivement $R_3^l = \prod_{i=1}^{l} C_3^i = \prod_{i=1}^{l} E_{PK}\ (r_i) = E_{PK}\left(\sum_{i=1}^{l}(r_i)\right)$) « l'agrégation » des chiffrés C_2^i (respectivement C_3^i). Seuls R_2^l et R_3^l sont conservés par l'urne privée.

Soit $B_{l+1} = (C_1^{l+1}, C_2^{l+1}, C_3^{l+1}, ZKP_1^{l+1}, ZKP_2^{l+1}, S_{l+1})$ un nouveau bulletin reçu par l'urne privée. Après avoir effectué les vérifications d'usage, l'urne privée va agréger C_2^{l+1} (respectivement C_3^{l+1}) à R_2^l (respectivement R_3^l) : $R_2 = C_2^{l+1} \times R_2^l = \prod_{i=1}^{l+1} C_2^i = \prod_{i=1}^{l+1} E_{PK}\ (v_i) = E_{PK}\left(\sum_{i=1}^{l+1}(v_i)\right)$ (respectivement $R_3 = C_3^{l+1} \times R_3^l = \prod_{i=1}^{l+1} C_3^i = \prod_{i=1}^{l+1} E_{PK}\ (r_i) = E_{PK}\left(\sum_{i=1}^{l+1}(r_i)\right)$). L'urne privée supprime B_{l+1} et ne conserve que R_2 et R_3.

A la clôture du scrutin, chaque membre M_i du bureau de vote va ensuite procéder au déchiffrement partiel du dernier chiffré R_2 (respectivement R_3) conservé par l'urne privée, à l'aide de sa clé privé SK_{M_i}. Dès qu'on a obtenu *s* (le seuil) déchiffrements partiels du message R_2, il est possible, en les combinant, de retrouver le texte clair $Res = \sum_{i=1}^{l+1}(v_i)$ (respectivement $Rand = \sum_{i=1}^{l+1}(r_i)$) qui correspond au nombre de voix qui se sont portées pour le « oui ». On obtient donc le résultat de ce référendum.

Les k membres (M_i) du bureau de vote publient Res et $Rand$. L'audit du scrutin se fait de manière analogue à la procédure décrite à la section 2.4.4.

Notre système de vote est ainsi le premier à « *garantir que l'identité de l'électeur ne pourra être mise en relation avec l'expression de son vote, et cela à tout moment du processus de vote, y compris après le dépouillement* » [Del10] tout en offrant « *la transparence de l'urne pour tous les électeurs à partir d'outils tiers* » [Del19].

3. REFERENCES :

[Ar07] Arrêté du 25 avril 2007 pris en application du décret n° 2007-602 du 25 avril 2007 relatif aux conditions et aux modalités de vote par voie électronique pour l'élection des délégués du personnel et des représentants du personnel au comité d'entreprise et modifiant le code du travail

[BD+17] Amira Barki, Nicolas Desmoulins, Saïd Gharout, Jacques Traoré: Anonymous attestations made practical. WISEC 2017: 87-98

[CFP+10] Benoît. Chevallier-Mames, Pierre-Alain Fouque, David Pointcheval, Julien Stern, Jacques Traoré : “On Some Incompatible Properties of Voting Schemes”, Towards Trustworthy Elections - New Directions in Electronic Voting. Lecture Notes in Computer Science 6000, Springer 2010, ISBN 978-3-642-12979-7, pages 191-199.

[CGS97] Cramer, R., Gennaro, R., Schoenmakers, B.: A secure and optimally efficient multi-authority election scheme. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 103–118. Springer, Heidelberg (1997)

[Cha81] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. Communications of the Association for Computing Machinery, 24(2) :84{88, 1981.

[CPP13] Edouard Cuvelier, Olivier Pereira, Thomas Peters: Election Verifiability or Ballot Privacy: Do We Need to Choose? ESORICS 2013: 481-498

[Dec07] Décret n° 2007-602 du 25 avril 2007 relatif aux conditions et aux modalités de vote par voie électronique pour l'élection des délégués du personnel et des représentants du personnel au comité d'entreprise et modifiant le code du travail.

[Del10] Délibération n° 2010-371 du 21 octobre 2010 portant adoption d'une recommandation relative à la sécurité des systèmes de vote électronique – CNIL

[Del19] Délibération n° 2019-053 du 25 avril 2019 portant adoption d'une recommandation relative à la sécurité des systèmes de vote par correspondance électronique, notamment via Internet – CNIL

[DGS12] Denise Demirel, Jeroen van de Graaf, Roberto Samarone dos Santos Araújo: Improving Helios with Everlasting Privacy Towards the Public. EVT/WOTE 2012

[Doc07] https://www.docapost.com/solutions/vote

[DSA98] Norme ANSI X9.62-1998, Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)

[Elg85] El Gamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 10–18. Springer, Heidelberg (1985)

[GMR85] Shafi Goldwasser, Silvio Micali and Charles Rackoff. The knowledge complexity of interactive proof-systems (extended abstract). In 17th ACM STOC, pages 291–304. ACM Press, May 1985.

[GNPR09] Ronen Gradwohl, Moni Naor, Benny Pinkas and Guy N. Rothblum: Cryptographic and Physical Zero-Knowledge Proof Systems for Solutions of Sudoku Puzzles. Theory Comput. Syst. 44(2): 245-268 (2009)

[JJR02] Markus Jakobsson, Ari Juels, Ronald L. Rivest: Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking. USENIX Security Symposium 2002: 339-353

[LL90] A. K. Lenstra and H. W. Lenstra, Jr. Algorithms in number theory. In J. van Leeuwen, editor, Handbook of Theoretical Computer Science, pages 673-715. Elsevier Science Publishers B.V., Amsterdam, 1990.

[Pai99] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In Jacques Stern, editor, EUROCRYPT'99, volume 1592 of LNCS, pages 223–238. Springer, Heidelberg, May 1999.

[Ped92] Torben P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In Joan Feigenbaum, editor, CRYPTO'91, volume 576 of LNCS, pages 129–140. Springer, Heidelberg, August 1992.

[Sch89] Claus-Peter Schnorr, « Efficient Identification and Signature for Smart Cards », Theory and Application of Cryptology, Springer, 1989

[Vox06] https://www.voxaly.com/solutions/vote-par-internet/