

Ce document est sous licence Creative Commons : [CC-BY-NC](#).

1 Introduction et Contexte

1.1 Le paradoxe du vote électronique

Le vote électronique représente l'un des défis les plus fascinants et les plus controversés de l'informatique moderne. Depuis les années 2000, de nombreux pays ont expérimenté ou déployé des systèmes de vote dématérialisé, motivés par des promesses séduisantes : réduction des coûts, rapidité du dépouillement, accessibilité pour les personnes à mobilité réduite, facilitation du vote à distance pour les expatriés, et modernisation de la démocratie à l'ère numérique.

Pourtant, contrairement à d'autres domaines où la numérisation s'est imposée comme une évidence (banque, commerce, administration), **le vote électronique reste profondément contesté par la communauté scientifique en sécurité informatique**. Ce paradoxe apparent révèle une vérité fondamentale : tous les problèmes ne peuvent pas être résolus par davantage de technologie.

1.2 Les exigences contradictoires du vote démocratique

Le vote dans une démocratie doit satisfaire simultanément un ensemble de propriétés qui, en système électronique, entrent en conflit mathématique et technique :

a) Authenticité et Unicité

Chaque électeur doit pouvoir voter une et une seule fois. Le système doit donc :

- Authentifier l'identité de l'électeur
- Enregistrer qu'il a exercé son droit de vote
- Empêcher tout vote multiple

b) Secret du vote et Anonymat

Le vote doit être secret et il doit être impossible de relier un bulletin à son auteur, y compris a posteriori. Cette exigence protège contre :

- La coercition (menaces, pressions)
- L'achat de votes
- Les représailles politiques

c) Transparence et Vérifiabilité

Le processus électoral doit être auditable et compréhensible par un citoyen ordinaire. Chacun doit pouvoir :

- Vérifier que son vote a été correctement enregistré
- Constater que le décompte est honnête
- Faire confiance au résultat sans compétence technique

d) Impossibilité de prouver son vote

Paradoxalement, tout en permettant la vérification individuelle, le système doit empêcher qu'un électeur puisse prouver à un tiers pour qui il a voté. Cette contrainte, souvent négligée, est cruciale pour prévenir :

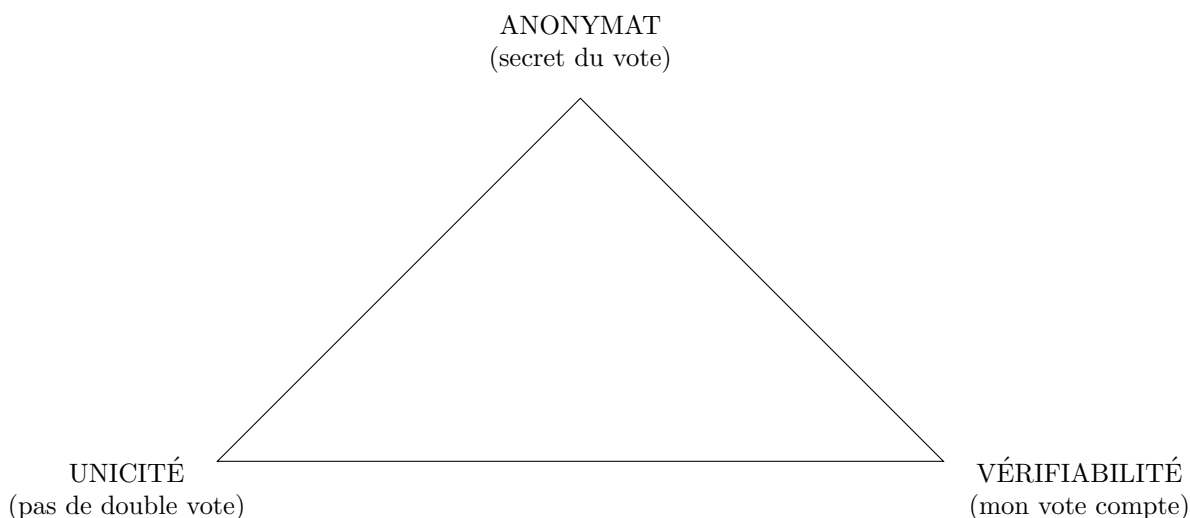
- La vente de votes (l'acheteur exige une preuve)
- Le vote sous contrainte (le coerciteur demande une capture d'écran)

e) Robustesse et Disponibilité

Le système doit fonctionner de manière fiable pendant toute la durée du scrutin, résister aux pannes, aux attaques, et permettre un recomptage en cas de contestation.

1.3 Le triangle d'impossibilité

Ces exigences forment ce que les cryptographes appellent parfois le "triangle d'impossibilité du vote électronique" :



Le problème fondamental : en système électronique, ces trois propriétés sont contradictoires dans un système électronique réaliste sans hypothèses de confiance irréalistes :

- Pour garantir l'unicité, il faut tracer qui a voté → menace sur l'anonymat
- Pour garantir l'anonymat, il faut déconnecter l'identité du bulletin → menace sur la vérifiabilité individuelle
- Pour garantir la vérifiabilité, il faut que l'électeur puisse prouver son vote → menace sur l'impossibilité de preuve (donc risque de coercition)

Note : Les systèmes End-to-End Verifiable (E2E-V) tentent de résoudre ce triangle en introduisant une quatrième dimension : la vérifiabilité sans révélation. Nous ne la traiterons pas dans ce document, vous pourrez toutefois en parler en conclusion.

1.4 Les limites intrinsèques de la sécurité informatique

Au-delà des contraintes démocratiques, le vote électronique se heurte à des réalités techniques fondamentales :

a) Le problème de la confiance centralisée

Dans un système de vote papier, la confiance est distribuée : scrutateurs de différents partis, observateurs, multiples intervenants. Pour frauder massivement, il faut corrompre des centaines de personnes dans des dizaines de bureaux.

Dans un système électronique, la confiance est centralisée : quelques développeurs, administrateurs système, ou une entreprise privée contrôlent l'intégralité du processus. Une seule ligne de code malveillante peut fausser toute une élection.

b) L'opacité du code

Un bulletin papier est compréhensible par tous. Un système informatique est une boîte noire :

- Le code source peut contenir des backdoors
- Même audité, le code peut être modifié après certification
- Le code qui tourne n'est pas nécessairement celui qui a été audité

c) Le terminal de l'électeur est un point de faille critique

Pour le vote par Internet, l'ordinateur ou smartphone de l'électeur devient un maillon faible :

- Malwares qui modifient le vote avant envoi
- Keyloggers qui volent les identifiants
- Screen capture pour prouver le vote (coercition)
- Aucun environnement contrôlé

d) L'impossibilité du recomptage indépendant

Les bulletins papier sont des objets physiques conservables et recomptables à l'infini par n'importe qui. Les votes électroniques sont des données numériques :

- Modifiables sans laisser de traces (avec accès privilégié)
- Dépendantes d'un logiciel pour être "lues"
- Impossibles à recompter sans refaire confiance au même système

1.5 Objectif de ce TP

Face à ces constats, ce TP ne vous demande pas de résoudre le problème du vote électronique (ce serait prétentieux, puisque les meilleurs cryptographes du monde n'y sont pas parvenus).

L'objectif est triple :

- Technique : Implémenter un système de vote électronique "basique" pour comprendre les mécanismes d'authentification, de chiffrement et de décompte.
- Analytique : Identifier les vulnérabilités, comprendre pourquoi chaque "amélioration" crée de nouveaux problèmes, et saisir les limites fondamentales de la sécurité informatique appliquée au vote.
- Citoyen : Développer un esprit critique sur les solutions technologiques présentées comme "sûres", comprendre que certains problèmes sociaux ne peuvent pas être résolus par la technique seule, et réfléchir aux compromis entre efficacité et démocratie.

Ce TP est autant un exercice de programmation qu'une leçon d'humilité technique et de responsabilité citoyenne.

2 Fonctionnement proposé

Ce TP reprend largement le TD d'Anca Nitulescu intitulé *La cryptographie appliquée au vote électronique*

2.1 Les intervenants

Pour organiser le scrutin, nous allons faire intervenir plusieurs entités, cela permettra en particulier de réduire le pouvoir de chacune. Ces entités sont :

- Le commissaire au vote qui servira à vérifier le droit de vote d'un citoyen ;
- L'administrateur qui servira à concevoir des bulletins de vote authentiques et infalsifiables ;
- L'anonymiseur qui réceptionnera les votes tel une urne ;
- Le décompteur qui comptera les bulletins une fois la session de vote finalisée.

L'administrateur et le décompteur disposent chacun d'un système cryptographique avec une clé publique et une clé privée.

2.2 Préparation du scrutin

A chaque électeur est envoyée une carte où figurent deux codes N_1 et N_2 générés aléatoirement. Pour fixer les idées, ces codes seront formés par 12 caractères, chiffres ou lettres, ce qui propose $(10 + 26)^{12} \approx 10^{18}$ combinaisons pour chacun des codes ressemblant à celui-ci :

AF15 GH25 8ZQP

Le commissaire au vote dispose de la liste de tous les codes N_1 valides. Pour une population de 1 million de citoyens, la probabilité de former au hasard un code N_1 valide est

$$\frac{1\,000\,000}{(10 + 26)^{12}} \approx 10^{-12}.$$

A l'aide d'une fonction de hachage cryptographiquement sûre (SHA-256 minimum), on transforme tous les codes N_2 en leur empreinte N'_2 .

On détruit ensuite la liste des codes N_2 , et on communique au commissaire au vote la liste des empreintes N'_2 . Le commissaire au vote ne connaît donc pas les codes N_2 , ceci l'empêchera par la suite de pouvoir introduire des bulletins frauduleux.

2.3 Déroulement du scrutin

La période de votation s'ouvre. De son ordinateur et à l'aide d'un logiciel de vote officiel, l'électeur contacte l'administrateur. Il lui transmet son code N_1 . L'administrateur vérifie auprès du commissaire au vote la validité du code N_1 puis, dans l'affirmative, approuve le droit de voter.

L'électeur fait son choix de vote et entre son code N_2 . Le logiciel de vote forme alors un message constitué de ce vote, du code N_2 et complété par des bits aléatoires : ceci est son bulletin de vote.

L'administrateur s'apprête à signer numériquement le vote de l'électeur. Cependant, il ne faut pas que l'administrateur ait connaissance de ce vote et c'est là que va intervenir le protocole de signature à l'aveugle.

Signature à l'aveugle

Nous allons proposer un protocole où Bob appose sa signature sur un message produit par Alice sans pour autant en connaître le contenu, comme si ce message figurait dans une enveloppe en papier qu'il lui suffit de signer pour apposer sa signature sur le document sans pour autant déchiffrer l'enveloppe.

Supposons que Bob dispose d'un système cryptographique RSA de clé publique (e, N) et de clé privée d . Alice souhaite voir Bob signer un message m codé sous la forme d'un entier compris entre 0 et $N-1$. Alice commence par choisir un entier k premier avec N qui joue le rôle de **facteur de masquage**. Alice transmet ensuite à Bob l'entier :

$$m' = mk^e \mod N$$

Ne connaissant pas k , Bob ne peut pas déterminer m et ne connaît donc pas le message transmis par Alice. Bob peut néanmoins apposer sa signature au message en transmettant à Alice la valeur :

$$m'' = (m')^d \mod N$$

À partir de cette signature masquée m'' , Alice n'a alors plus qu'à évaluer :

$$s = m'' \cdot k^{-1} \mod N$$

pour disposer d'un entier s vérifiant $s^d = m \mod N$, avec k^{-1} l'inverse modulaire de k modulo N . Ainsi, avec le couple (m, s) , Alice dispose d'un message signé par Bob alors que celui-ci ne connaît pas la nature du message qu'il a indirectement signé.

Note de vérification : Pour vérifier que s est bien la signature de m par Bob, n'importe qui peut calculer $s^e \mod N$ et vérifier que le résultat égale m . C'est le principe de la signature RSA classique qui est préservé ici.

Par un facteur de masquage, l'électeur transforme son vote et en demande la signature à l'administrateur.

Une fois cette signature reçue, il retire le facteur de masquage et il dispose désormais d'un bulletin de vote authentifié par l'administrateur. Il n'y a plus qu'à déposer le bulletin dans l'urne.

Le logiciel de vote contacte alors l'anonymiseur et lui envoie le code N_1 ainsi que le vote signé, ce dernier étant préalablement chiffré par la clé publique du décompteur, ce qui revient en fait à mettre le vote dans une enveloppe.

L'anonymiseur vérifie auprès du commissaire que le code N_1 est valide et, dans l'affirmative, le commissaire raye le code N_1 de la liste des codes valides et l'anonymiseur enregistre le vote.

2.4 Dépouillement

La période de votation est close, l'urne est pleine, il n'y a plus qu'à dépouiller, c'est le rôle du décompteur.

À l'aide de sa clé privée, le décompteur déchiffre tous les bulletins transmis par l'anonymiseur, ce qui revient à retirer les bulletins de leur enveloppe.

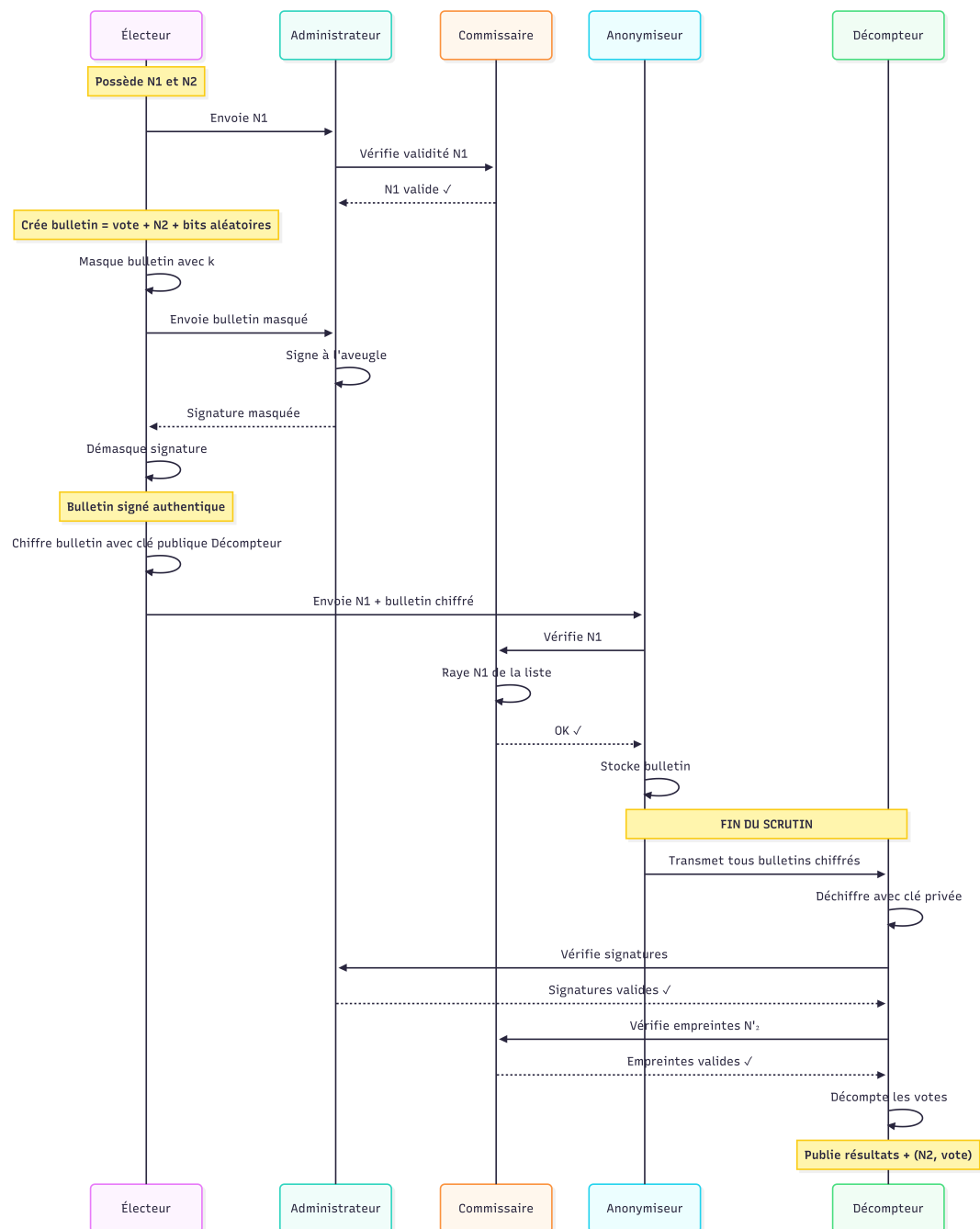
Le décompteur réalise ensuite deux vérifications pour chaque bulletin :

- Il vérifie l'authenticité de leur signature en exploitant la clé publique de l'administrateur,
- Il transmet ensuite au commissaire le code N_2 figurant sur le bulletin et celui-ci calcule alors l'empreinte du code N_2 afin de vérifier que celle-ci figure dans sa liste des empreintes N'_2 .

Une fois ces vérifications effectuées, le vote est décompté.

3

Récapitulatif et Sécurité du protocole



Dans ce protocole, sans disposer du code N_1 , personne ne peut usurper le droit de vote d'un citoyen. Aussi, sans disposer du code N_2 , personne ne peut enregistrer un vote valide. Ainsi, ce protocole peut être considéré comme sûr au niveau des intervenants extérieurs. Il reste à vérifier que les quatre entités (administrateur, commissaire, anonymiseur et décompteur) ne peuvent pas fausser la procédure :

- **Le commissaire** : Celui-ci a le pouvoir d'accepter ou de refuser des votes car c'est lui qui possède la liste des codes N_1 valides. Cependant, il n'est jamais en contact direct ni avec le votant, ni avec le vote. De plus, ne connaissant les codes N_2 que par leur empreinte, il est dans l'impossibilité de concevoir des votes valides et ne peut donc pas remplir l'urne.
- **L'administrateur** prend connaissance lui aussi des codes N_1 valides mais il n'a pas non plus accès aux codes N_2 , ce qui l'empêche de remplir l'urne. L'administrateur est en contact avec le votant mais lorsqu'il signe son vote, il n'en connaît pas la nature car celui-ci a été préalablement masqué. De plus, il lui est à terme impossible de relier le véritable vote au votant.

- **L'anonymiseur** a aussi connaissance du code N_1 mais pas du code N_2 . Puisque le vote transmis est chiffré par la clé publique du décompteur, il ne connaît pas le contenu du vote. Même après le scrutin, lorsque les couples formés par N_2 et le vote associé sont publiés, il ne peut faire le lien avec un vote chiffré qui lui aurait été transmis à cause des bits aléatoires qui ont été adjoints pour former le bulletin de vote.
- **Le décompteur** a connaissance du vote mais ne peut le relier au votant. Il a aussi connaissance du code N_2 et pourrait donc transformer le vote introduit à sa guise. Cependant, la période de votation est close et l'administrateur ne signe alors plus aucun vote, ce qui empêche le décompteur de concevoir un vote authentique.

4

Questions

Question 1

Ces dernières années, on peut trouver de nombreuses situations où le vote électronique a été contesté, et où des fraudes ou des risques importants de fraudes ont été relevés.

Donnez des éléments sourcés sur ces situations.

Question 2

De nombreux organismes internationaux, voire des personnalités de premier plan, ont émis des avis concernant le vote électronique.

Donnez des éléments sourcés sur ces avis.

Question 3

Codez en Python, PHP ou NodeJS une application WEB qui implémente le protocole décrit ci-dessus. Voici quelques points à prendre en compte :

- Fichier CSV ou JSON contenant des électeurs fictifs
- Implémentation propre des protocoles, des entités
- Implémentation des entités sous la forme d'une classe
- Aucune entité ne doit avoir accès à des données qui ne lui sont pas destinées

Question 4

On peut imaginer des scénarios d'attaque de ce protocole :

- **Attaques par les entités internes** : Que se passe-t-il si deux des quatre entités (commissaire, administrateur, anonymiseur, décompteur) collaborent pour frauder ? Identifiez les paires les plus dangereuses.
- **Administrateur malveillant** : Bien que l'administrateur ne voie pas le contenu des votes, peut-il les altérer d'une manière ou d'une autre ? Comment ?
- **Décompteur corrompu** : Le décompteur a accès aux bulletins déchiffrés. Quelles manipulations peut-il faire ? Quelles sont ses limites ?
- **Attaque de l'anonymiseur** : Un pirate compromet le serveur de l'anonymiseur pendant le scrutin. Quelles données peut-il voler ? Peut-il désanonymiser les votes ?

Question 5

Après avoir implémenté ce protocole et analysé ses vulnérabilités, rédigez une synthèse (1 à 2 pages) abordant les points suivants :

- **Comparaison avec le vote papier** : Ce protocole offre-t-il les mêmes garanties qu'un vote papier traditionnel ? Quelles propriétés démocratiques sont affaiblies ? Lesquelles sont améliorées (le cas échéant) ?
- **Confiance vs Transparence** : Ce système repose-t-il sur plus de confiance ou plus de transparence que le vote papier ?
- **Positionnement personnel** : Seriez-vous favorable au déploiement d'un tel système pour une élection présidentielle ? Dans quels contextes (élections locales, référendums, votes d'entreprise, etc.) ce système serait-il acceptable selon vous ?
- **Améliorations futures** : Y a-t-il des technologies émergentes (blockchain, zero-knowledge proofs, chiffrement homomorphe, etc.) qui pourraient résoudre certaines limites identifiées ?