Preuve à divulgation nulle de connaissance pour la vérification d'éligibilité dans le vote par internet

Benjamin Voisin

28 août 2023

Le vote électronique

Les propriétés importantes

Privacy

Garantir le secret du vote, de manière robuste dans le temps

Vérifiabilité

Permettre de vérifier le bon déroulement de l'élection :

- Vérification du calcul des résultats
- Vérification l'intégrité de l'urne publique
- **Vérification de l'éligibilité des votants**

Le problème de l'authentification

Preuve d'éligibilité

Il ne faut pas juste s'authentifier auprès du serveur de vote, il faut pouvoir prouver aux autres que notre bulletin correspond à un votant éligible. On veut donc fournir une preuve d'éligibilité.

Distribution des identifiants

Phase critique du vote : Il faut se protéger du vol et de la vente d'identifiants.
On peut utiliser des identifiants déjà existant (France Connect, par exemple).

Notre protocole d'authentification

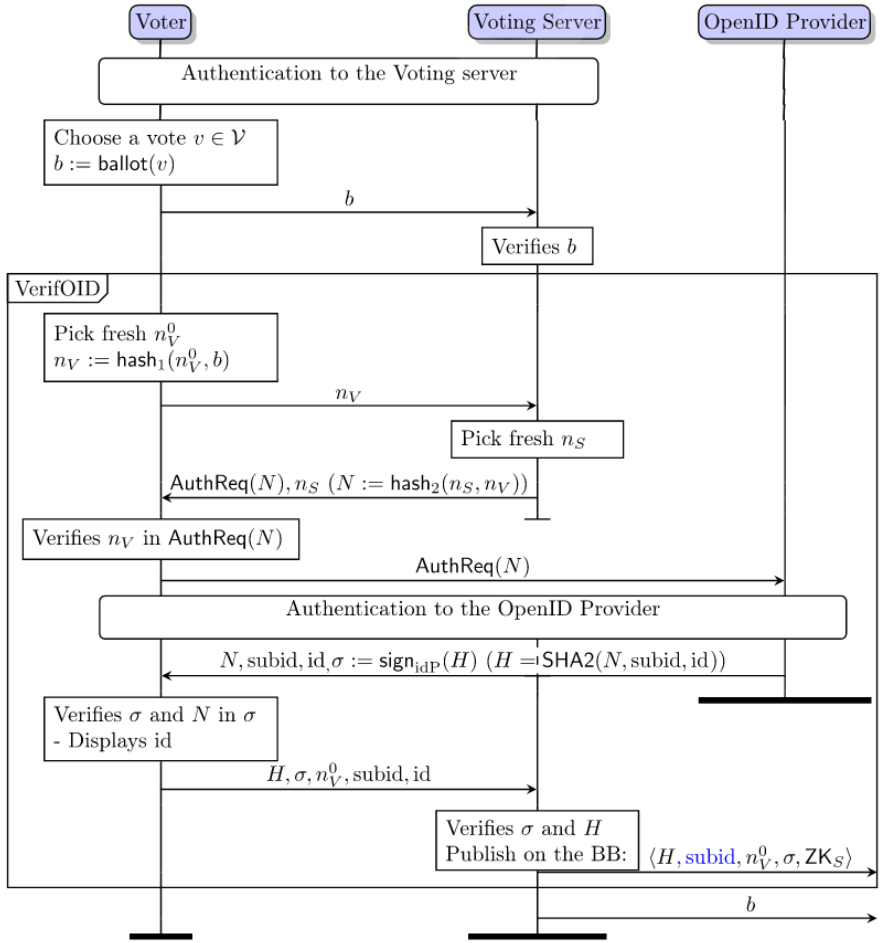

Figure – Schéma du protocole d'authentification avec OpenID Connect

Preuves à divulgation nulle de connaissance (ZKP)

Prouver l'éligibilité sans révéler l'identité

On peut prouver des propriétés sur un élément, tout en le gardant secret. Par exemple, prouver qu'on connait la solution d'une grille de sudoku, sans révéler le remplissage. On peut ainsi résoudre tous les problèmes NP.

Preuve non interactive

On peut simuler l'interactivité pour rendre une preuve non interactive : Le prouveur génère une preuve qu'il publie, et n'importe qui à n'importe quel moment peut vérifier que la preuve est correcte, sans avoir à interagir avec le prouveur.

Description du circuit

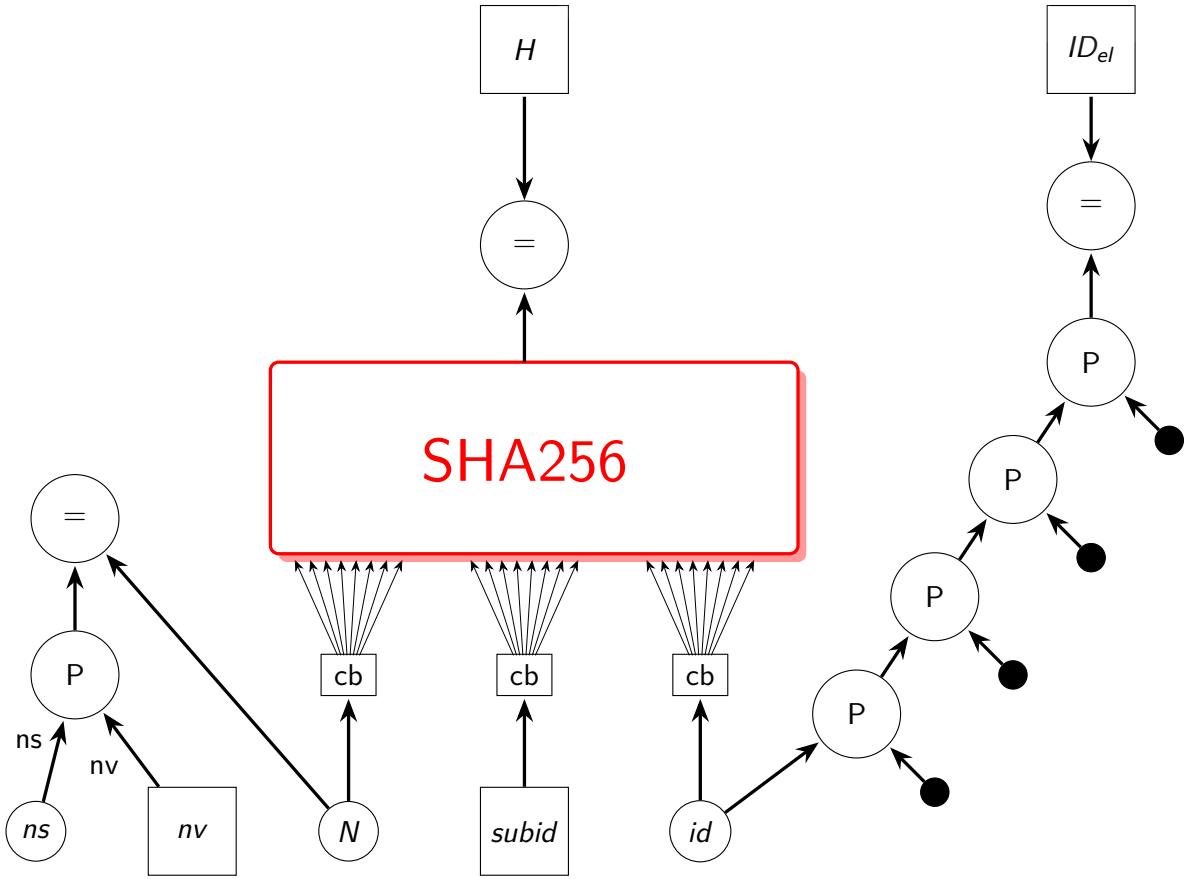

Figure – Schéma du circuit arithmétique de la preuve ZK_S

Conversion base64

Le protocole OpenID Connect oblige à faire une conversion base64 avant le hash. Il faut donc l'ajouter au circuit de preuve.

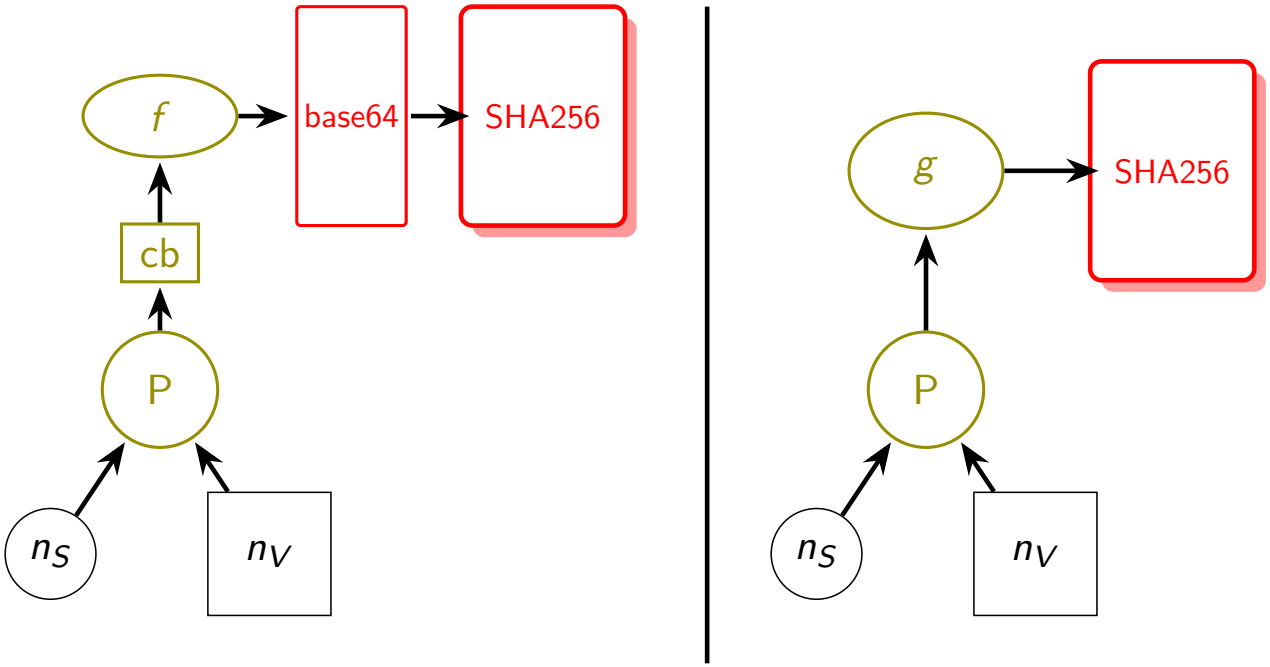

Figure – Schéma du circuit de conversion en base64

Circuit g

On ajoute "00" devant chaque demi-mot de 4 bits, pour donner un mot de 6 bits représenté par une lettre entre "A" et "P" en base 64.

Extrait de la table ASCII

bin	0000	0001	0010	0011	0100	...	1111
0000	NUL	SOH	STX	ETX	EOT	...	SI
...							
0011	0	1	2	3	4	...	?
0100	@	A	B	C	D	...	O

Extrait de la table base64

000000	A
000001	B
000010	C
000011	D
...	
001111	P

Évaluation

Résultats temporels

Sur une machine de 16 cœurs physique et 500GB de RAM :

- Temps de génération de preuve : 6.5 secondes
- Temps de génération de preuve + construction circuit : 20 secondes
- Temps de vérification : 10 ms
- Temps de vérification + construction circuit : 10 secondes

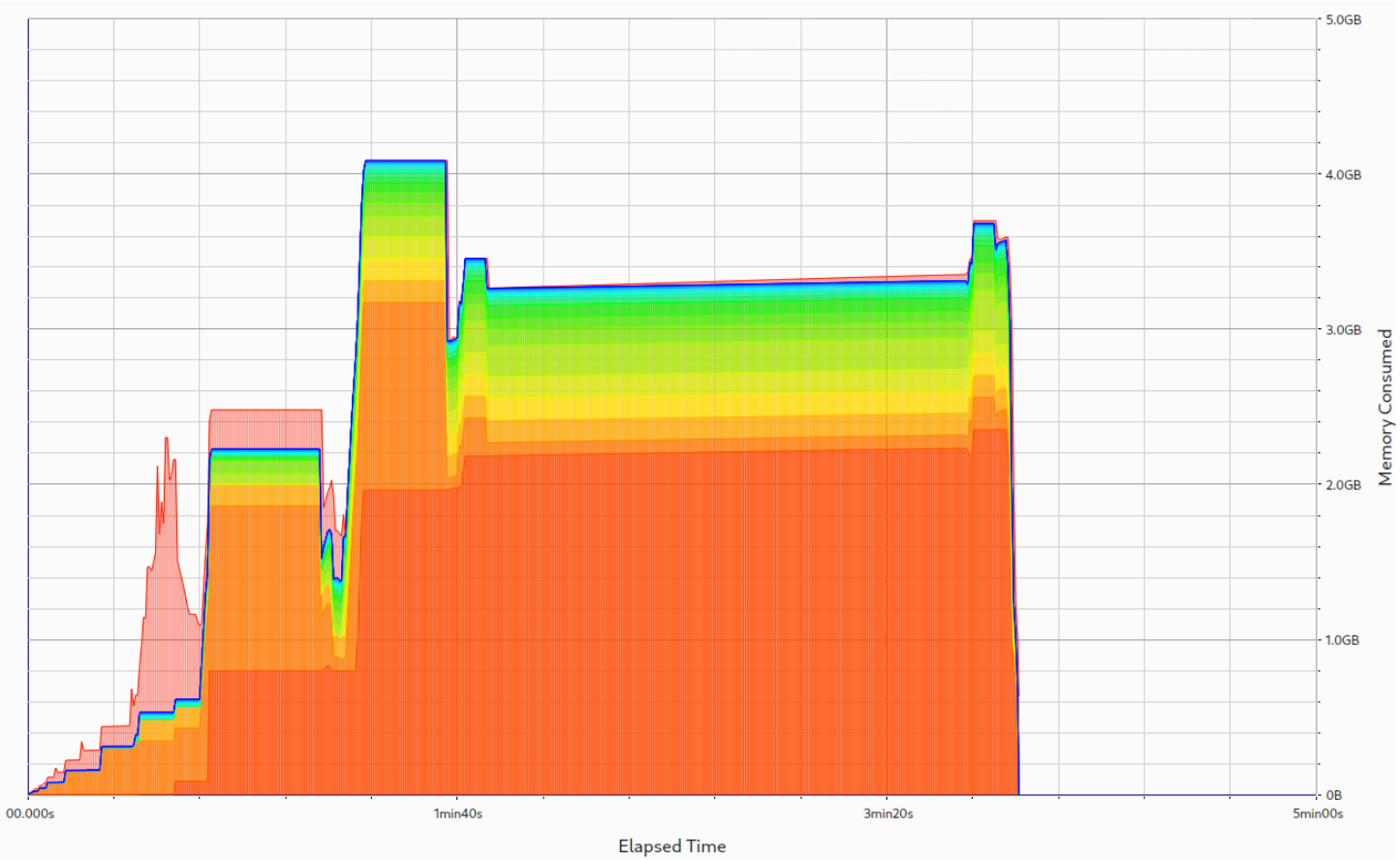

Figure – Utilisation de la RAM pour la génération d'une preuve

Conclusion

Faisabilité en pratique

5h30 de génération de preuve pour 1 000 votants, et 55h pour 10 000.
En réutilisant le circuit de preuve : 1h48 pour 1 000 votants, et 18h pour 10 000.

Axes d'amélioration

- Rendre le circuit réutilisable
- Utiliser Starky pour la preuve de hash SHA256
- Générer la preuve sur l'appareil du votant

Bibliographie

Véronique Cortier, Pierrick Gaudry, and Stéphane Glondu.

Belenios : A Simple Private and Verifiable Electronic Voting System, pages 214–238.
Springer International Publishing, Cham, 2019.

Alexandre Debant and Lucca Hirschi.

Reversing, breaking, and fixing the french legislative election e-voting protocol.
Cryptology ePrint Archive, Paper 2022/1653, 2022.

Amos Fiat and Adi Shamir.

How to Prove Yourself : Practical Solutions to Identification and Signature Problems, page 186–194.
Springer-Verlag, Berlin, Heidelberg, 1987.

Oded Goldreich, Silvio Micali, and Avi Wigderson.

Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems.
J. ACM, 38(3):690–728, jul 1991.

S Goldwasser, S Micali, and C Rackoff.

The Knowledge Complexity of Interactive Proof-Systems, page 291–304.
STOC '85. Association for Computing Machinery, New York, NY, USA, 1985.

Lorenzo Grassi, Dmitry Khovratovich, Christian Rechberger, Arnab Roy, and Markus Schofnegger.

Poseidon : A new hash function for Zero-Knowledge proof systems.
In *30th USENIX Security Symposium (USENIX Security 21)*, pages 519–535. USENIX Association, August 2021.