Everlasting privacy dans le vote électronique

Rapport de stage encadré par Alexandre Debant et Lucca Hirschi au Loria

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**

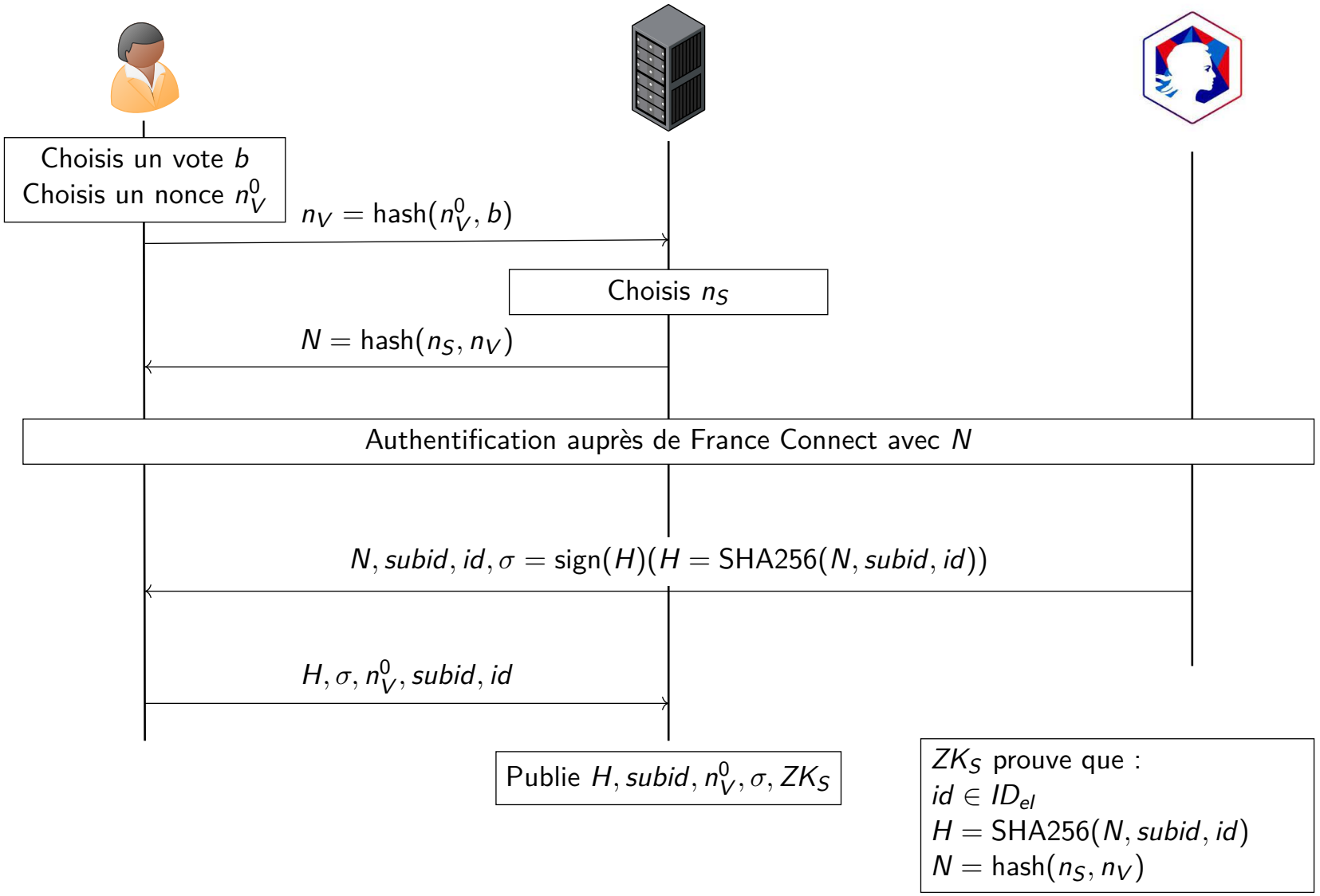
Le problème de l'authentification

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).


Choisis un vote b
Choisis un nonce n_V^0
$n_V = \text{hash}(n_V^0, b)$
Choisis n_S
$N = \text{hash}(n_S, n_V)$
Authentification auprès de France Connect avec N
$N, subid, id, \sigma = \text{sign}(H)(H = \text{SHA256}(N, subid, id))$
$H, \sigma, n_V^0, subid, id$
Publie $H, subid, n_V^0, \sigma, ZK_S$
ZK_S prouve que :
$id \in ID_{el}$
$H = \text{SHA256}(N, subid, id)$
$N = \text{hash}(n_S, n_V)$

Preuves à divulgation nulle de connaissance (ZKP)

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

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

Pauline

Victor

Preuves à divulgation nulle de connaissance (ZKP)

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

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

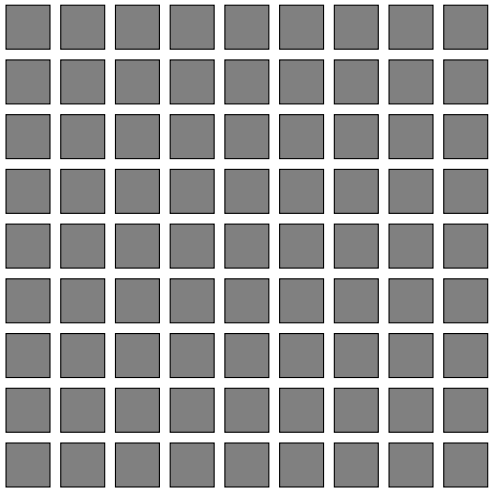
Pauline

Victor

Preuves à divulgation nulle de connaissance (ZKP)

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

Pauline

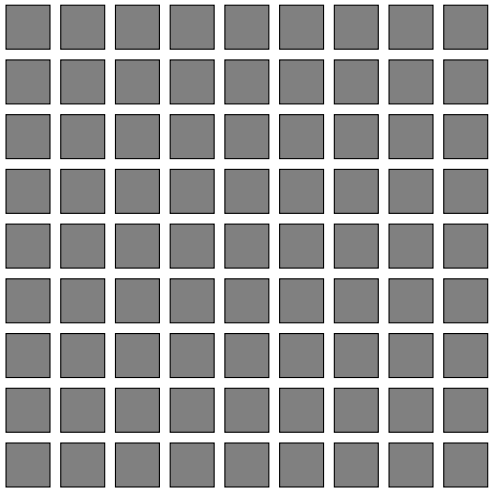


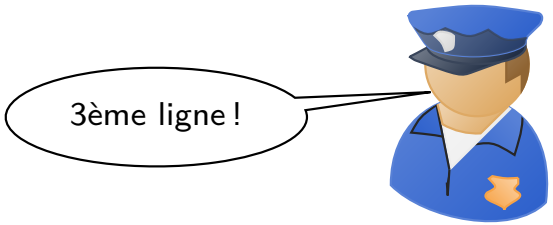
Victor

Preuves à divulgation nulle de connaissance (ZKP)

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

Pauline

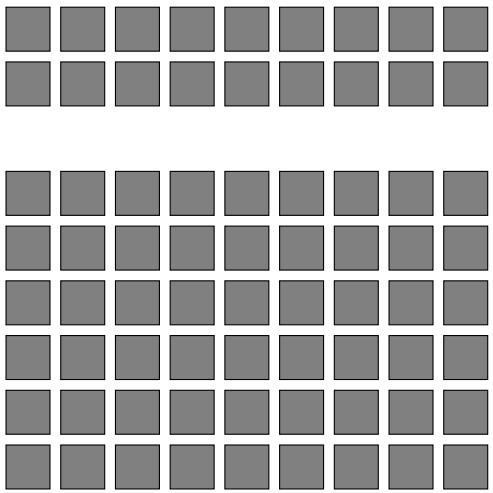




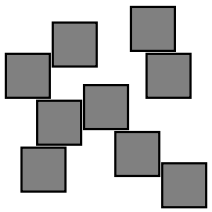
Victor

Preuves à divulgation nulle de connaissance (ZKP)

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



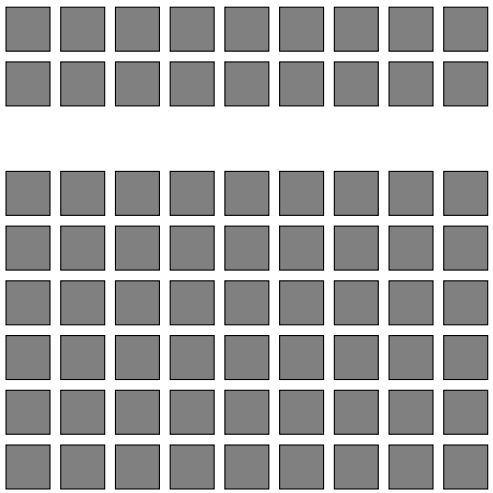
Pauline



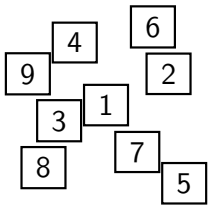
Victor

Preuves à divulgation nulle de connaissance (ZKP)

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



Pauline

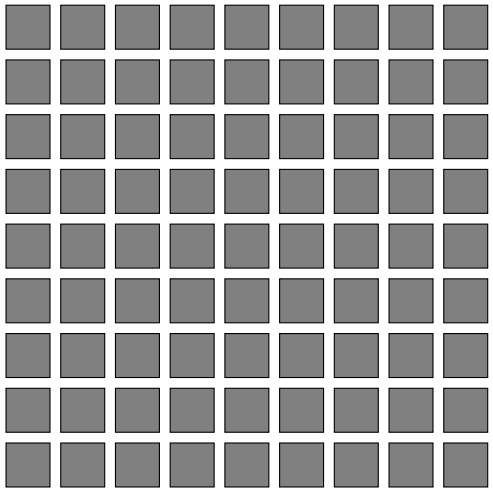


Victor

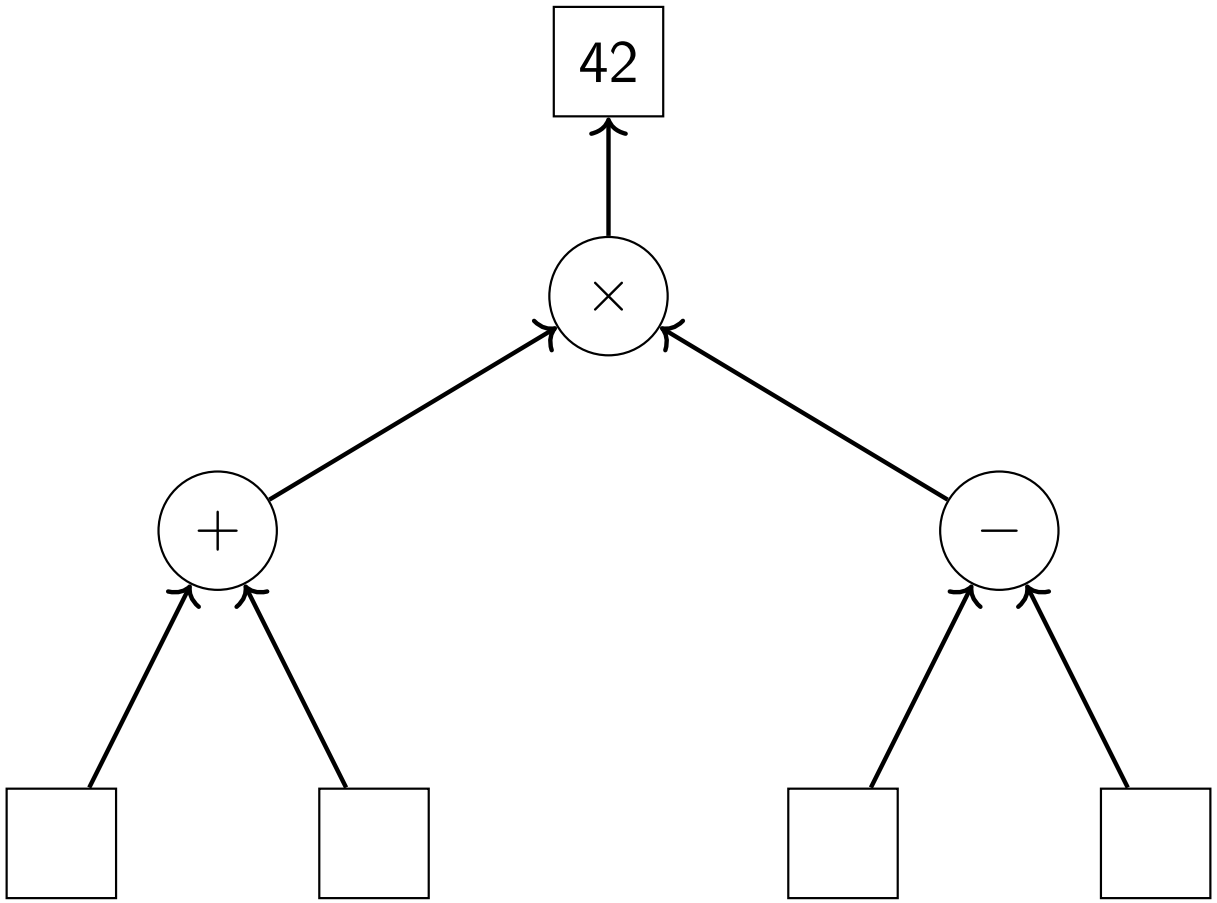
Preuves à divulgation nulle de connaissance (ZKP)

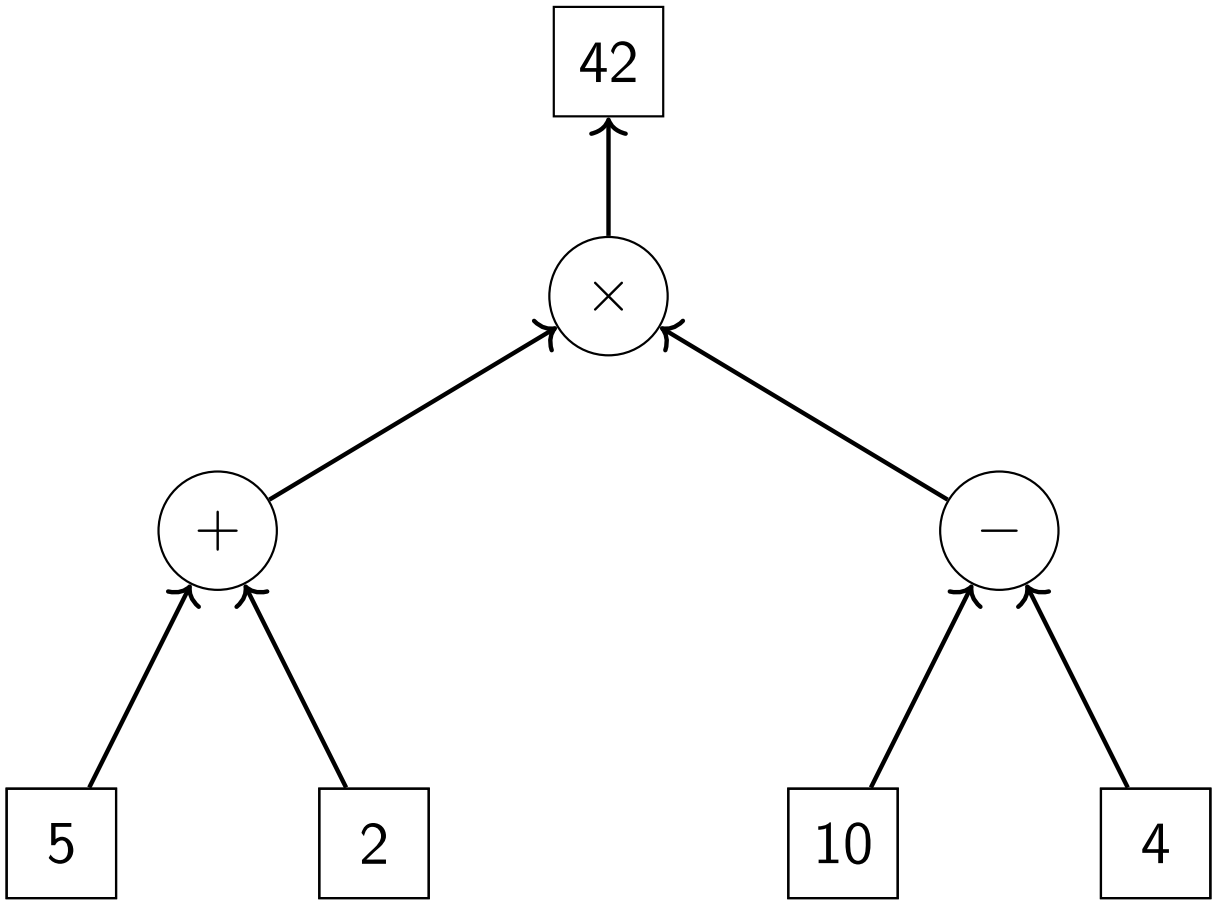
	2		5		1		9	
8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

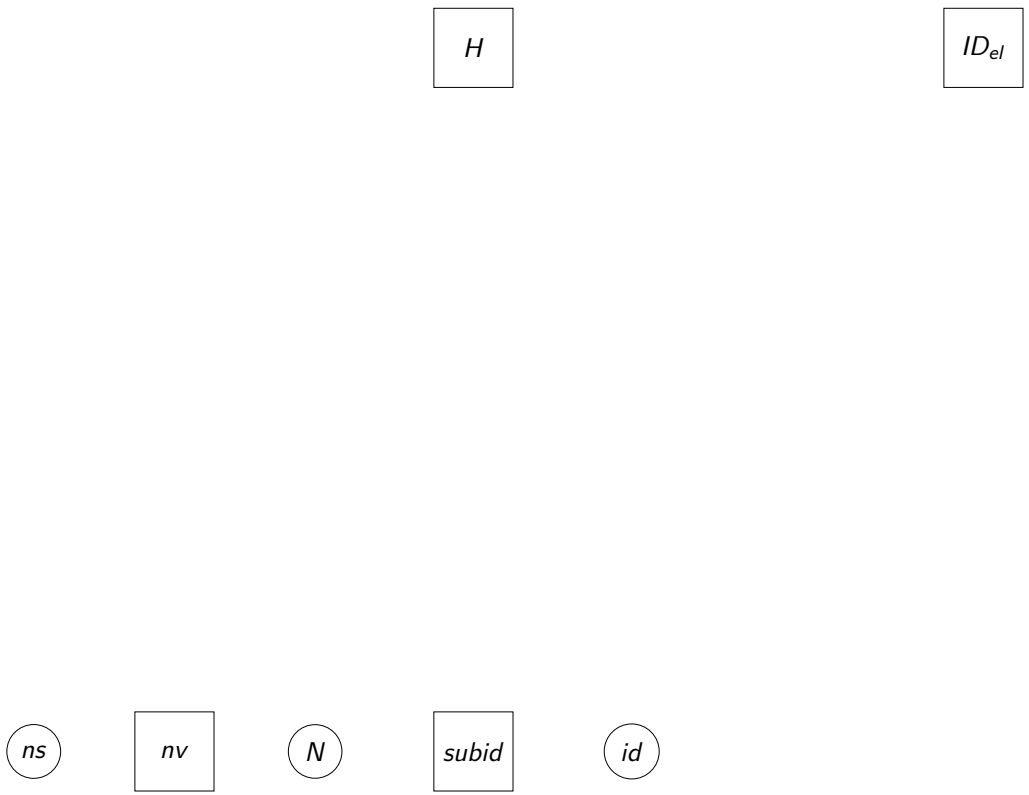
Pauline



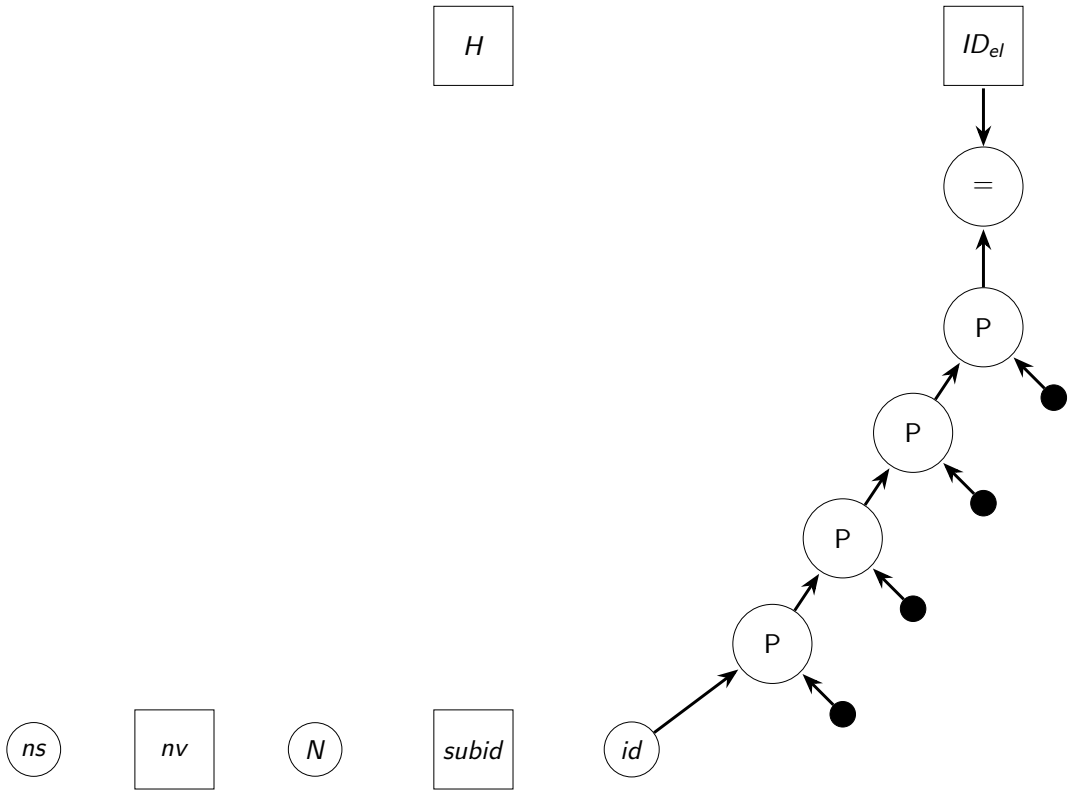
Victor


42
×
+
−

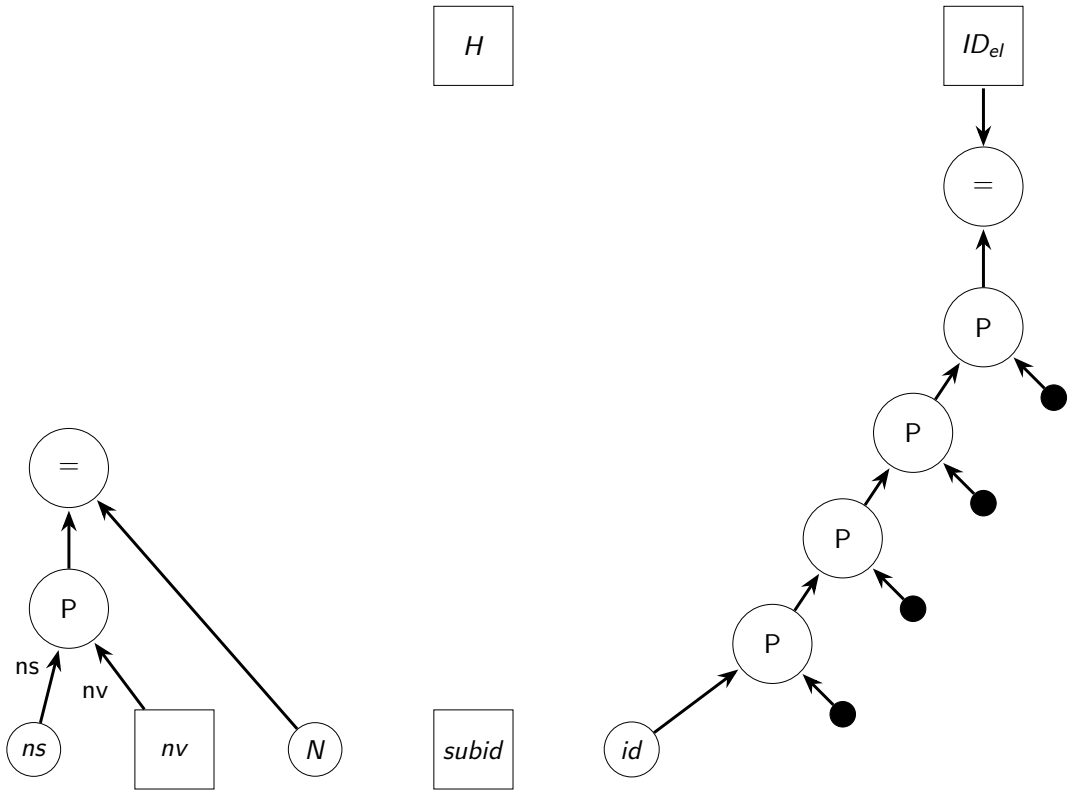

42
×
+
−
5
2
10
4


H
ID_{el}
ns
nv
N
subid
id

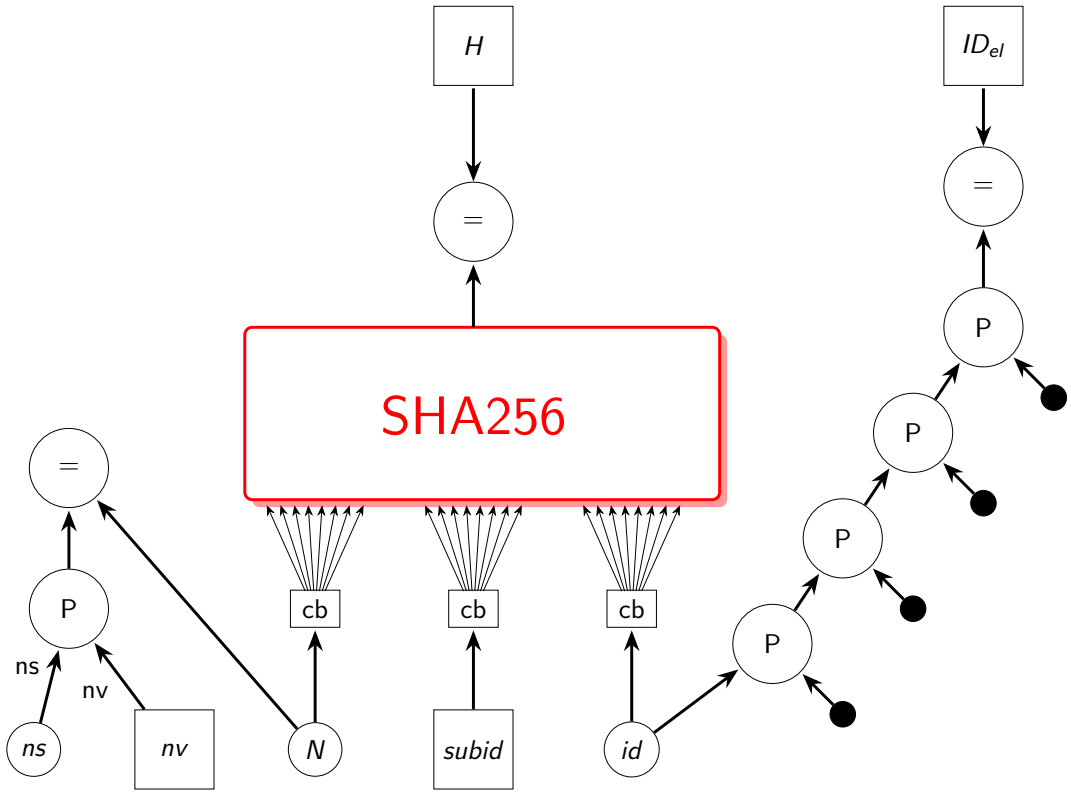
- $id \in ID_{el}$



- $id \in ID_{el}$
- $N = \mathsf{hash}(n_S, n_V)$

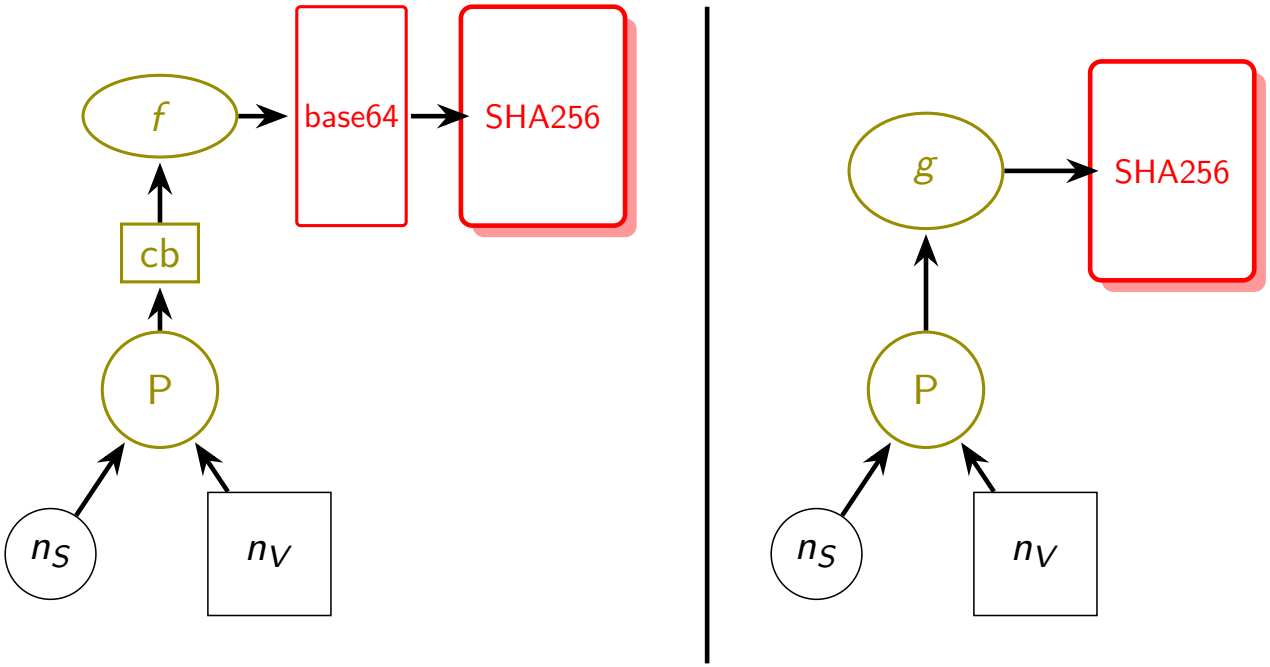


- $id \in ID_{el}$
- $N = \text{hash}(n_S, n_V)$
- $H = \text{SHA256}(N, subid, id)$



Conversion base64

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



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

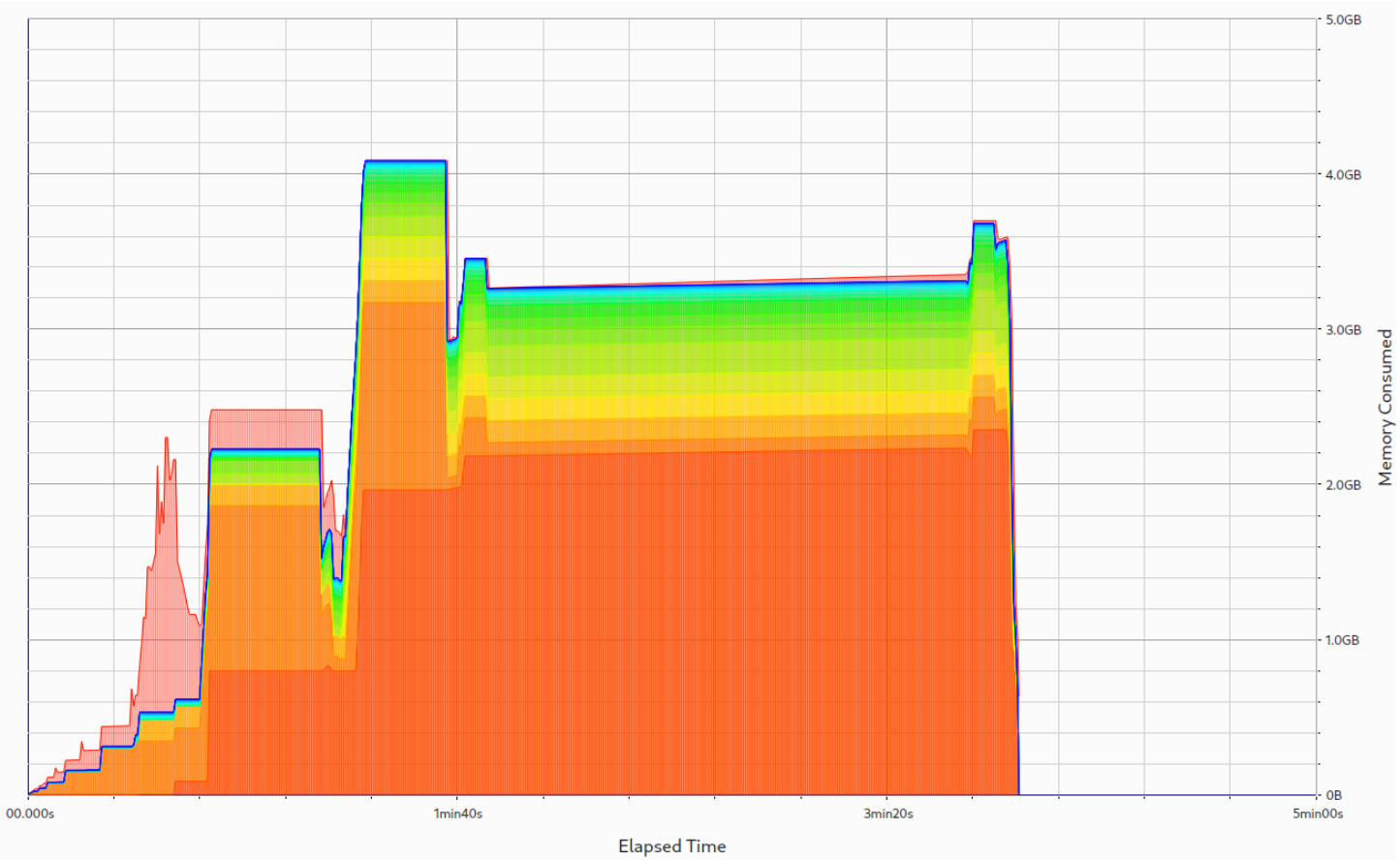
000000	A
000001	B
000010	C
000011	D
...	
001111	P

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


5.0GB
4.0GB
3.0GB
2.0GB
1.0GB
0B
Memory Consumed
00.000s
1min40s
3min20s
5min00s
Elapsed Time

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