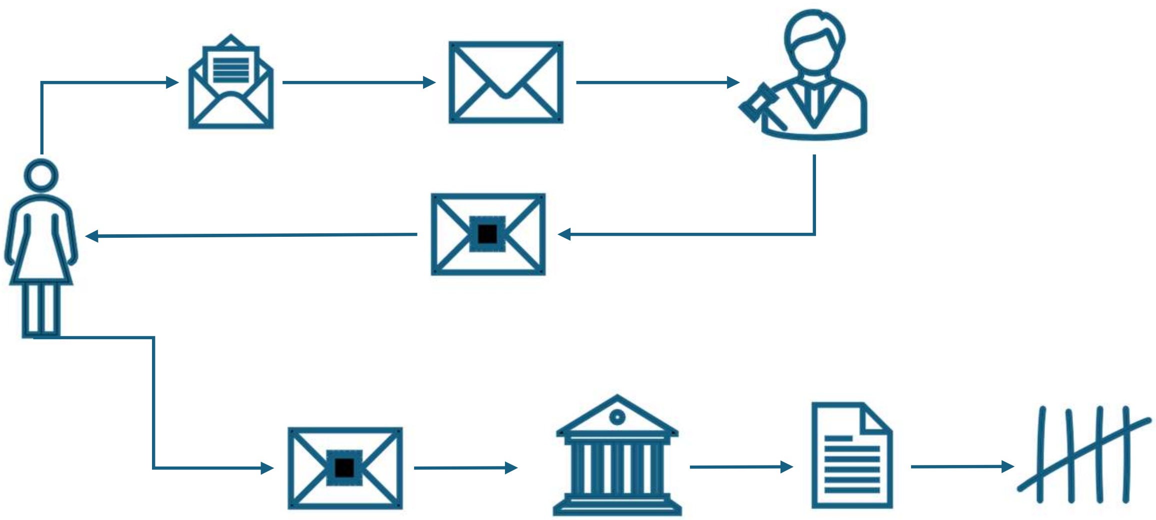
Voting With Blind Signatures

A practical secret voting scheme for large scale elections

- **Main idea:** How would real-world elections work if the identity validation took place in a different physical space than counting?



Assumptions:

- Voters have cryptographic key pairs
- Voters can send two messages
- Access to an anonymous channel

Fujioko A, Okamoto T, Ohta T (1992) A practical secret voting scheme for large-scale elections. In: Proceedings of advances in cryptology, AUSCRYPT’92, Springer, pp 244–260

A practical secret voting scheme for large scale elections (FOO)

- Preparation $\mathrm{V_i}$
  - Select and commit to the vote
    - $b_i = Commit(v_i, r_i)$
  - Blind ballot for pk_{EA}
    - $bb_i = Blind(b_i, pk_{EA})$
  - Sign with voter sk_{v_i}
    - $sbb_i = Sign(sk_{v_i}, bb_i)$
  - Send to the EA
    - (id, bb_i, sbb_i)

- Authorisation by the EA
  - Check voter eligibility and previous authorization requests for double voting
  - If everything is ok sign the blind ballot and return it to the voter
    - $sbb_i{}' = Sign(sk_{EA}, bb_i)$
  - Announce total number of eligible voters by publishing to the BB
    - $T = \{id, bb_i, sbb_i{}'\}$

A practical secret voting scheme for large scale elections (FOO)

- Voting **Phase 1**
  - Unblind the ballot signature
    - $sb_i = Unblind(sbb_i')$
  - Send ballot and signature to the BB through **an anonymous channel**
    - (b_i, sb_i)
  - Eligibility is publicly verifiable by verifying the EA signature
  - Everybody can create a list of eligible ballots and verify it against T
- Voting **Phase 2**
  - **After everyone has voted!**
  - Send decommitment values over an anonymous channel
    - (v_i, r_i)
- **(Public)** Counting Phase
  - Verify all commitments
  - Verify eligibility
  - Compute tally using successfully verified decommited values

A practical secret voting scheme for large scale elections (FOO)

- Voting **Phase 1**
  - Unblind the ballot signature
    - $sb_i = Unblind(sbb_i')$
  - Send ballot and signature to the BB through **an anonymous channel**
    - (b_i, sb_i)
  - Eligibility is publicly verifiable by verifying the EA signature
  - Everybody can create a list of eligible ballots and verify it against T
- Voting **Phase 2**
  - **After everyone has voted!**
  - Send decommitment values over an anonymous channel
    - (v_i, r_i)
- **(Public)** Counting Phase
  - Verify all commitments
  - Verify eligibility
  - Compute tally using successfully verified decommited values

Voting with Blind Signatures: Discussion

- **Privacy**
  - Commitment schemes
  - Blind signatures
  - Anonymous channel
- Major difference with Helios
  - **There is no need to require trust in the server for privacy!**

- **Verifiability**
  - Individual
    - Existence of the signed ballots and decommitments in the BB
  - Universal
    - Counting can be replicated
    - No secret keys involved
  - Elibigility
    - Based on the unforgeability of the blind signature scheme

But: Voting is a two-step process in different protocol phases

Voting With Ring Signatures

LSAG Voting

- Remove the authority from the FOO scheme
- All voters have cryptographic key pairs
- There is a **reliable** repository of identities and public key pairs
  - Who creates it?
- **Voting phase:**
  - Each voter picks v_i and signs it using a LSAG scheme
    - The ring is selected from the public repository of identities
  - The ballot is (v_i, σ_i)
  - The ballot is posted via an anonymous channel
- **Tallying phase:**
  - Everyone can retrieve the ballots from the BB and verify the signatures by retrieving the identities from the repository

Joseph K. Liu, Victor K. Wei, and Duncan S. Wong. "Linkable Spontaneous Anonymous Group Signature for Ad Hoc Groups (Extended Abstract)". In: *ACISP 2004*. Vol. 3108. LNCS. 2004, pp. 325–335. doi: 10. 1007/978-3-540-27800-9_28.

LSAG Voting

- **Privacy**
  - Anonymous channel
  - Ring anonymity
- **Verifiability**
  - The counting process can be performed by everyone
  - The linkability property of the LSAG prevents double voting

Joseph K. Liu, Victor K. Wei, and Duncan S. Wong. “Linkable Spontaneous Anonymous Group Signature for Ad Hoc Groups (Extended Abstract)”. In: *ACISP 2004*. Vol. 3108. LNCS. 2004, pp. 325–335. doi: 10. 1007/978-3-540-27800-9_28.

Open Vote Network

Decentralised Voting

A different paradigm

- Large scale election
  - Authorities involved
    - mixing,
    - tallying,
    - BB maintenance
  - Some trust required
  - Each voter is only interested in casting their ballot
    - Vote & Go

- Boardroom voting
  - No entity is special
  - Conducted by the voters themselves
    - They may send other messages except their votes
  - Private channels lead to disputes
  - Robustness is important
    - A voter should not disrupt an election

Anonymous Voting by 2-Round Public Discussion

- **Setup**
  - Select a group $\mathbb{G}$ of prime order q
- **Preparation**
  - Each of n voters V_i samples $x_i \overset{\$}{\leftarrow} \mathbb{Z}_q$
- **Commitment**
  - Each V_i broadcasts $\langle g^{x_i}, DLPRV(x_i, g, g^{x_i})\rangle$
  - When every voter is finished everyone computes
    - $Y_i = \frac{\prod_{j=1}^{i-1} g^{x_j}}{\prod_{j=i+1}^{n} g^{x_j}} = g^{y_i}$ for some $y_i \in \mathbb{Z}_q$
- **Voting**
  - Each V_i selects $v_i \in \{0,1\}$ and broadcasts
    - $Y_i^{x_i} g^{v_i}$
- **Self-Tallying**
  - Everyone computes
    - $\prod_{i=1}^{n} Y_i^{x_i} g^{v_i} = g^{\sum_i v_i}$
  - Solve simple DLOG

Hao, Feng, Peter Y. A. Ryan and Piotr Zielinski. "Anonymous voting by two-round public discussion." *IET Inf. Secur.* 4 (2010): 62-67.

Protocol Magic

- Correctness
  - $\sum_i x_i y_i = \sum_{i=1}^{n} \sum_{j=1}^{i-1} x_i x_j - \sum_{i=1}^{n} \sum_{j=i+1}^{n} x_i x_j = 0$
- Intuition

	x_1	x_2	x_3	x_4
x_1		-	-	-
x_2	+		-	-
x_3	+	+		-
x_4	+	+	+	

Robustness - Fairness

- The protocol is not robust
  - If a voter that has participated in the **commitment round** does not participate in the **voting round** the result cannot be computed
- The protocol is not fair
  - The last voter learns the result before the rest
  - They can adapt their vote for a favorable result
- Solution: A recovery round

Recovery Round

- L: The set of voters that have performed both rounds
  - They participate in one more round in order to post cancellation tokens
  - They compute $Z_i \leftarrow \frac{\prod_{j\in[i+1,n]\setminus L} g^{x_i}}{\prod_{j\in[1,i-1]\setminus L} g^{x_i}}$
  - The cancellation token is $(Z_i^{x_i}, \boldsymbol{DLPRV}(x_i, g, Z_i))$
  - They are used to remove the commitments of the players that did not vote
- Tallying becomes
  - $\prod_{i=1}^{n} Y_i^{x_i} g^{v_i} \cdot \prod_{i\in L} Z_i^{x_i} = g^{\sum_{i\in L} v_i}$

No	First round	Second round	Third round	Recovery
1	g^{x_1}	commitment	$g^{x_1 y_1} = g^{x_1(-x_2-x_3-x_4-x_5)}$	$\hat{h}_1^{x_1} = g^{x_1(x_2+x_4)}$
2	g^{x_2}	commitment	Abort	–
3	g^{x_3}	commitment	$g^{x_3 y_3} = g^{x_3(x_1+x_2-x_4-x_5)}$	$\hat{h}_3^{x_3} = g^{x_3(x_4-x_2)}$
4	g^{x_4}	commitment	Abort	–
5	g^{x_5}	commitment	$g^{x_5 y_5} = g^{x_5(x_1+x_2+x_3+x_4)}$	$\hat{h}_5^{x_5} = g^{x_5(-x_2-x_4)}$

Implementation on the Blockchain

- Ethereum
- Smart Contracts for
  - Registration (using the accounts of the voter)
  - Voting
  - Tallying
- Restrictions
  - integers of 256 bits
  - expensive cryptographic computations
  - one vote or six registrations per block
  - small number of allowed local variables
  - order of transactions in a block and timers
- Linear Complexity for Tally And Vote
- Maximum number of voters: 50
- Cost/voter: 0.73$ (2017)

Patrick McCorry, Siamak Shahandashti, and Feng Hao, A smart contract for boardroom voting with maximum voter privacy, pp. 357–375, 01 2017.

Improvements

- Organize voters in Merkle Tree
  - only the root is stored (256 bits)
- Instead of voter list a voter provides a proof of membership
- Tally off-chain by an untrusted tallier
- Publish computation trace in Merkle Tree
- Subject to verification

Mohamed Seifelnasr and Hisham Galal, Scalable open-vote network on ethereum, pp. 436–450, 08 2020

Voting on the blockchain

- Conceptual similarity between blockchain and the BB
  - Append-only
  - Broadcast channel
  - No central authority - anyone can be a miner (given enough computing power)
  - Pseudonymity
- Good for universal/individual verifiability (recorded as cast)
- But...
- Registration/authentication/eligibility verifiability are inherently centralized
- Does not help with verifying voter intent
- Does not help with coercion-resistance / receipt-freeness
- Intensifies threats associated with everlasting privacy
- Is it actually decentralized? (concentration of mining power)

Voting on the blockchain

- To sum up... ‘*using Blockchain for voting solves a small part of the problem with an unnecessarily big hammer*’ (Ben Adida, 2017)
- However...
- ...it might be useful for different types of elections
  - new election paradigms on a smaller scale
  - blockchain governance