opn.vote: A Publicly Verifiable Blockchain-Based eVoting System

Felix Maduakor[1], Thi Van Thao Doan[2], and Joerg Mitzlaff[1]

[1] openPetition gGmbH, Germany
[2] Université catholique de Louvain, B-1348 Louvain-la-Neuve, Belgium
maduakorfelix@gmail.com, thi.doan@uclouvain.be, joerg@openpetition.net

1 Motivation

Verifiability in e-voting systems is commonly achieved through a public bulletin board [1, 5, 12], typically managed by a centralized entity such as a voting server [4]. This centralized approach not only introduces vulnerabilities from the single point of failure (e.g., denial-of-service attacks, data breaches) but can also undermine core security guarantees, such as privacy, correctness, and resistance to voter suppression. Public blockchains offer a promising solution by shifting critical e-voting processes to a decentralized network. This ensures protocol integrity and correct execution without reliance on a single authority, mitigating risks like censorship, tampering, or system failure. However, many blockchain-based e-voting protocols overlook practical constraints, including limited computational resources and transaction costs [8]. When deployed, such protocols often require voters to install wallet software and acquire cryptocurrency to pay transaction fees, significantly hindering accessibility.

Contributions. We introduce opn.vote, a decentralized and publicly verifiable e-voting system designed to eliminate the single point of failure in centralized approaches. By leveraging Ethereum's EIP-4337 (Account Abstraction) [3], opn.vote removes the need for voters to install wallets or hold cryptocurrency, significantly improving accessibility. Moreover, opn.vote reduces transaction costs by $\approx 51\%$ compared to MACI [9], the leading voting system on Ethereum.

The design of opn.vote draws inspiration from FOO's approach [7], in which, after interacting with an authority, each voter derives an anonymous credential and casts an anonymous vote. To formally capture voter privacy, we introduce a cryptographic notion of *voter anonymity* based on indistinguishability, ensuring that even a coalition of all participating authorities cannot link a voter to their cast vote, assuming an anonymous channel. opn.vote achieves voter anonymity through Schnorr blind signatures and guarantees both universal and individual verifiability by publishing the decryption key after tallying, enabling straightforward public verification. Furthermore, unlike FOO, where each voter can cast only one vote, opn.vote permits vote recasting without requiring re-registration.

From a practical perspective, opn.vote will be deployed during the ABSTIMMUNG21 referendum in Germany in September 2025. The system aims to transition 50% of paper votes to online voting, reducing the costs per vote from €2 to under €0.01.

2 System Design

In opn.vote, the public bulletin board is realized through the Ethereum blockchain (see Figure 1). Each voter V holds a key pair $(\mathsf{sk_V}, \mathsf{vk_V})$, where $\mathsf{sk_V}$ is a randomly generated signing key that remains private to V, and $\mathsf{vk_V}$ is the verification key. Each ballot submission is represented as an ERC-4337 UserOperation (UserOp), which is digitally signed using $\mathsf{sk_V}$. $\mathsf{vk_V}$ is unique to each voter and also serves as their smart wallet address. Together, $(\mathsf{sk_V}, \mathsf{vk_V})$ form the voter's smart wallet.

Election Setup. The Election Coordinator CO generates an ElGamal key pair $(\mathsf{ek}, \mathsf{dk})$ [6] and publishes the encryption key ek within the Voting Smart Contract VSC. The Registrar R creates a signing key pair $(\mathsf{sk_R}, \mathsf{vk_R})$ and publishes $\mathsf{vk_R}$ in the VSC. Election metadata is stored in the VSC and on IPFS [2].

Registration Phase. After eligibility verification by CO, each V interacts with R to obtain a valid signature cred, which is R's signature on $\mathsf{vk_V}$. In opn.vote, this process employs the Schnorr blind signature scheme [10,11]. The pair $(\mathsf{vk_V}, \mathsf{cred})$ forms the voter's anonymous credential. R publishes all successful registrations, and CO publishes all successful eligibility checks within the VSC.

Voting Phase. V creates a ballot $\mathsf{b} = (\mathsf{cred}, c)$, where c is the encryption of their vote v using CO's ek. Since $\mathsf{vk_V}$ matches V's wallet address, VSC extracts it directly from the transaction, eliminating the need for inclusion in the ballot. b is signed by $\mathsf{sk_V}$ and sent as a UserOp to the decentralized Bundler Network BN. Nodes in BN validate the UserOp against CO's sponsorship rules and submit an on-chain transaction invoking the VSC 's vote casting function. Transaction fees are sponsored by the CO through a Sponsor Smart Contract. To recast a vote, V generates a new ballot $\mathsf{b}' = (\varnothing, c')$, where $\varnothing$ is an empty credential. After the initial ballot, authentication uses the public key $\mathsf{vk_V}$, removing the need for cred in recasts. Only the most recent vote cast by each voter is counted.

Tallying Phase. After the election deadline, everyone can check whether the ballot is signed properly using $\mathsf{vk_V}$ and whether cred is a valid credential under $\mathsf{vk_R}$. Invalid ballots are discarded. CO then publishes the decryption key dk and the election results. Voters can verify that their ballot was correctly recorded and counted, and confirm the election results by running the tally process locally.

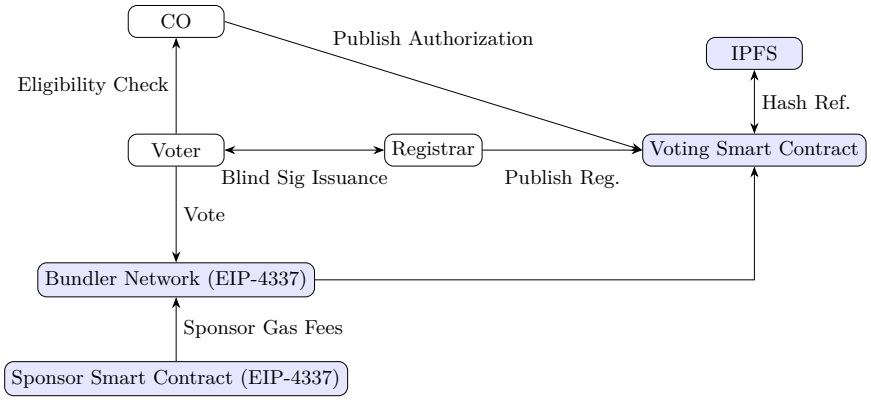

Fig. 1. High-level architecture of opn.vote, highlighting decentralized components.

References

1. Adida, B.: Helios: Web-based open-audit voting. In: Proceedings of the 17th USENIX Security Symposium. pp. 335–348. USENIX Association (2008)
2. Benet, J.: Ipfs-content addressed, versioned, p2p file system. arXiv preprint arXiv:1407.3561 (2014)
3. Buterin, V., Weiss, Y., Tirosh, D., Nacson, S., Forshtat, A., Gazso, K., Hess, T.: Erc-4337: Account abstraction using alt mempool. Tech. rep., Ethereum Improvement Proposals (2021)
4. Cortier, V., Lallemand, J., Warinschi, B.: Fifty shades of ballot privacy: Privacy against a malicious board. In: 2020 IEEE 33rd Computer Security Foundations Symposium (CSF). pp. 17–32. IEEE (2020)
5. Culnane, C., Ryan, P.Y.A., Schneider, S.A., Teague, V.: vvote: A verifiable voting system. ACM Trans. Inf. Syst. Secur. **18**(1), 3:1–3:30 (2015)
6. ElGamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory **31**(4), 469–472 (1985)
7. Fujioka, A., Okamoto, T., Ohta, K.: A practical secret voting scheme for large scale elections. In: Advances in Cryptology – AUSCRYPT'92. pp. 244–251. Springer (1993)
8. Kharman, A.M., Smyth, B.: Perils of current dao governance. arXiv preprint arXiv:2406.08605 (2024)
9. MACI: Maci github page (2022), https://github.com/privacy-scaling-explorations/maci, last accessed on 2023-11-17
10. Pointcheval, D., Stern, J.: Provably secure blind signature schemes. In: International Conference on the Theory and Application of Cryptology and Information Security. pp. 252–265. Springer (1996)
11. Pointcheval, D., Stern, J.: Security arguments for digital signatures and blind signatures. Journal of cryptology **13**, 361–396 (2000)
12. Ryan, P.Y., Rønne, P.B., Iovino, V.: Selene: Voting with transparent verifiability and coercion-mitigation. In: Financial Cryptography and Data Security: FC 2016 Workshops, pp. 176–192. Springer (2016)