FASTEN: Fair and Secure Distributed Voting UsingSmart Contracts

by

Sankarshan Damle, Sujit Prakash Gujar, Moin Hussain Moti

Report No: IIIT/TR/2021/-1

Centre for Others
International Institute of Information Technology
Hyderabad - 500 032, INDIA
May 2021

FASTEN: Fair and Secure Distributed Voting Using Smart Contracts

Sankarshan Damle and Sujit Gujar
Machine Learning Laboratory
International Institute of Information Technology (IIIT-H)
Hyderabad, India
sankarshan.damle@research.iiit.ac.in; sujit.gujar@iiit.ac.in

Moin Hussain Moti
The Hong Kong University of Science and Technology
Honk Kong
mhmoti@cse.ust.hk

***Abstract*—Electing democratic representatives via voting has been a common mechanism since the 17th century. However, these mechanisms raise concerns about fairness, privacy, vote concealment, fair calculations of tally, and proxies voting on their behalf for the voters. Ballot voting, and in recent times, electronic voting via electronic voting machines (EVMs) improves fairness by relying on centralized trust. Homomorphic encryption-based voting protocols also assure fairness but cannot scale to large scale elections such as presidential elections. In this paper, we leverage the blockchain technology of distributing trust to propose a smart contract-based protocol, namely, FASTEN. There are many existing protocols for voting using smart contracts. We observe that these either are not scalable or leak the vote tally during the voting stage, i.e., do not provide vote concealment. In contrast, we show that FASTEN preserves voter's privacy ensures vote concealment, immutability, and avoids double voting. We prove that the probability of privacy breaches is negligibly small. Further, our cost analysis of executing FASTEN over Ethereum is comparable to most of the existing cost of elections.**

***Index Terms*—Distributed Trust, Blockchain, Smart Contracts, Ethereum**

I. Introduction

Elections are fundamental to democratic governance. Since direct democracy - a form of government in which political decisions are made directly by the entire body of qualified citizens - is impractical, societies select governance representatives. Consequently, elections have been a common mechanism for modern representative democracy since 17th century [1]. They make it possible to include every eligible individual in the decision-making process by registering its vote into the system. A *fair election* is possible only when the voter can freely vote for its desired preference.

One must also ensure an agent's participation in the voting process is hidden. This can be achieved by eliminating the *link* between the voter and its vote, i.e., *anonymous* voting. To design such a fair election with anonymous voting, i.e., fair and secure election (FSE), we first define the following essential properties.

1) **Voter Anonymity (VA).** A vote cannot be traced back to the voter either during or after the election.
2) **Vote Concealment (VC).** The vote's value should remain hidden from the system (voters, candidate, election commission). VC, in turn, ensures that the vote tally remains a mystery to the system until the voting window has expired.
3) **Vote Immutable (VI).** Once a voter casts its vote, it should be impossible to alter it to any other vote by anyone.
4) **Double Voting Inhibition (DVI).** A voter should be allowed to vote only once in a specific election.

We thank Ripple for generous funding to establish Blockchain COE.

978-0-7381-1420-0/21/$31.00 ©2021 IEEE

FSE Overview. Towards FSE, the most traditional voting method is *paper ballots*. It partially ensures anonymity, vote concealment, and vote immutability. The major drawback of ballot-based voting is that it involves tiresome manual work in counting the votes. Along with the risk of unintentional and intentional human-error involved, the non-durability of paper and lack of a robust mechanism to avoid double voting are some of the other challenges involved with this system.

Election through *electronic voting machines* (EVM)s is a technological upgrade over the paper ballot system. EVMs provide voter anonymity that does not take voter ID as a parameter. Nevertheless, it fails at guaranteeing vote immutability. This failure is because the voter needs to trust the company that provides the EVM software for vote concealment and vote immutability. They also entrust companies with shipping EVMs with the correct version of the firmware, and thus, the EVM remains a black box to the voter. Besides, the double voting inhibition problem is still there.

Micali *et al.* [2] propose a protocol for secure auctions that is also applicable for voting. However, the body conducting the elections, *election commission* (EC), will know all the votes post the voting stage. To overcome the "centralized trust" placed on EC, the authors propose an expensive zero-knowledge proof of the result. Consequently, the protocol is also not scalable. Adida [3] proposes a most popular scheme *Helios*, which relies on the security of one server and is not viable for nationwide elections. Thus, there is a need to look for a completely different approach to conduct an FSE.

FSE over Blockchain. With Ethereum [4], we observe that blockchain can not only be used to solve the problem of designing a cryptocurrency but can also be used to implement *smart contracts* [4]. A smart contract allows blockchain to establish an interactive platform for n parties. Such a contract

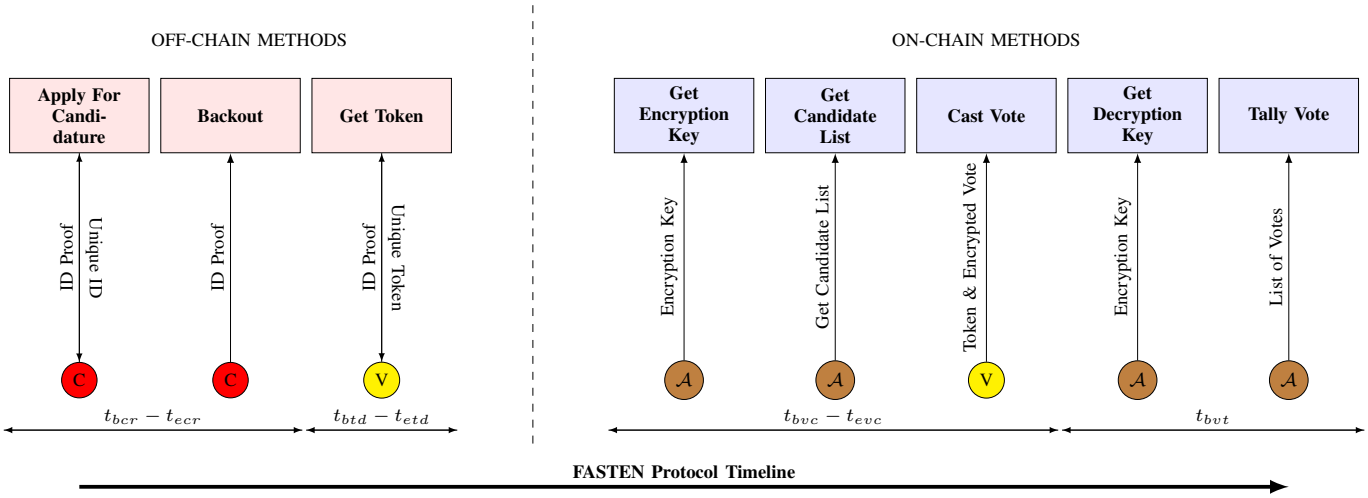

Fig. 1. Illustration of the protocol timeline in FASTEN. Here, C: Candidate, V: Voter, and $\mathcal{A}$: Other agent

enforces the outcome of any event through a set of rules. These rules correspond to a programming language understandable to the execution system. The key concept here is distributing trust rather than relying on a single party. Thus, we explore ways to leverage such a distributed trust to conduct an FSE.

The important steps in the voting procedure for FSE are: (i) *voter authentication*, i.e., a person claiming to be a voter should be an eligible voter; (ii) *vote registration*, which preserves the privacy of the voter as well as its vote; (iii) *outcome verification* that counts the tally of votes in a verifiable manner. Zhao *et al.* [5] propose a voting protocol based on *Commit-Publish* mechanisms that also leverages smart contract. As the authors' main goal is boardroom voting, it does not address step (i). It solves steps (ii) and (iii). To the best of our knowledge, in a plethora of voting schemes over blockchain, except [6], no protocol satisfies all the four desirable properties of a fair election. The challenge with [6] is scalability as it is Zcash-based, which is *considerably* slower than Bitcoin.

Our Approach. We propose a novel protocol for FSE, namely, FASTEN: *FAir and Secure disTributEd votiNg*. We partially rely on EC for authentication, which issues a random but *unique* token for each voter after authenticating the voter. The token is unique to the voter and the particular election. If the voter tries to obtain multiple tokens for the same election - it will receive the same token. Therefore, FASTEN is resistant to *Sybil* attacks. This authentication is similar to several secure applications over blockchain (e.g., [7]–[9]). In this work, we assume that EC does not store the link between a voter and its token[1]. Next, the smart contract considers tokens as an eligibility to vote granted by EC. After this, each voter registers its encrypted vote and token to the smart contract. The smart contract holds the hashes of all the tokens that EC issues. It computes the hash of the token registered by the voter and checks if it is present in the database. Once the entry is confirmed, the encrypted votes are registered in the voter database. After the vote casting window expires, the smart contract decrypts all the votes and computes the tally.

[1]Such trusted third-party authentications also have a close parallel with the Zcash Parameter Generation [10]. These links thus correspond to "toxic waste" – to be destroyed.

Notation	Definition
t_{bcr}	time-stamp for beginning the candidature registration
t_{ecr}	time-stamp for ending the candidature registration
t_{btd}	time-stamp for beginning the token distribution
t_{etd}	time-stamp for ending the token distribution
t_{bvc}	time-stamp for beginning the vote casting
t_{evc}	time-stamp for ending the vote casting
t_{bvt}	time-stamp for beginning the vote tally

TABLE I
TIME CONSTRAINTS IN FASTEN

Since our protocol deploys smart contracts based on blockchain, one may also implement it as a *Decentralized Application* (DApp). DApps comprise a friendly UI for any smart contract, thereby allowing layperson to interact with said contracts. Thus, FASTEN helps improve fairness in elections, i.e., designing FSE and improving voter participation.

II. FASTEN: PROTOCOL

We refer to the procedural methods of the overall protocol which are on the smart contract as *on-chain methods* and the remaining procedures as *off-chain methods*. Figure 1 illustrates FASTEN. The relevant notations are provided in Table I. We present the formal protocol in [11].

III. FASTEN: MAIN RESULT

Theorem 1. *FASTEN achieves fairness and security in election, i.e., FASTEN is a solution for FSE.*

Proof. Since FASTEN satisfies VA, VC, VI and DVI, it is a solution for FSE. □

REFERENCES

[1] R. G. Heinz Eulau, Paul David Webb *et al.*, "Election," 2015. [Online]. Available: https://www.britannica.com/topic/election-political-science

[2] S. Micali and M. O. Rabin, "Cryptography miracles, secure auctions, matching problem verification," *Communications of the ACM*, vol. 57, no. 2, pp. 85–93, 2014.

[3] B. Adida, "Helios: Web-based open-audit voting." in *USENIX security symposium*, vol. 17, 2008, pp. 335–348.

[4] V. Buterin, "Ethereum: A next-generation smart contract and decentralized application platform," *URL https://github. com/ethereum/wiki/wiki/% 5BEnglish% 5D-White-Paper*, 2014.

[5] Z. Zhao and T.-H. H. Chan, "How to vote privately using bitcoin," in *International Conference on Information and Communications Security*. Springer, 2015, pp. 82–96.

[6] B. Yu, J. K. Liu, A. Sakzad, S. Nepal, R. Steinfeld, P. Rimba, and M. H. Au, "Platform-independent secure blockchain-based voting system," in *International Conference on Information Security*. Springer, 2018, pp. 369–386.

[7] Y. Lu, Q. Tang, and G. Wang, "Zebralancer: Private and anonymous crowdsourcing system atop open blockchain," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2018, pp. 853–865.

[8] H. Duan, Y. Zheng, Y. Du, A. Zhou, C. Wang, and M. H. Au, "Aggregating crowd wisdom via blockchain: A private, correct, and robust realization," in *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. IEEE, 2019, pp. 1–10.

[9] S. Damle, B. Faltings, and S. Gujar, "A truthful, privacy-preserving, approximately efficient combinatorial auction for single-minded bidders," in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS '19, Montreal, QC, Canada, May 13-17, 2019*, E. Elkind, M. Veloso, N. Agmon, and M. E. Taylor, Eds. International Foundation for Autonomous Agents and Multiagent Systems, 2019, pp. 1916–1918. [Online]. Available: http://dl.acm.org/citation.cfm?id=3331962

[10] "Parameter generation - zcash." [Online]. Available: https://z.cash/ko_KR/technology/paramgen/

[11] S. Damle, S. Gujar, and M. H. Moti, "FASTEN: fair and secure distributed voting using smart contracts," *CoRR*, vol. abs/2102.10594, 2021. [Online]. Available: https://arxiv.org/abs/2102.10594