Votereum: An Ethereum-based E-voting system

Linh Vo-Cao-Thuy, Khoi Cao-Minh, Chuong Dang-Le-Bao, Tuan A. Nguyen
Faculty of Computer Networks and Communications
University of Information Technology
Vietnam National University HCMC, Vietnam
{14520473, 14520432}@gm.uit.edu.vn
{chuongdlb, tuanna}@uit.edu.vn

***Abstract*—Blockchain, which is the underlying technology of the first cryptocurrency Bitcoin, has drawn a lot of global attention in recent years. Its notable characteristics of the distributed ledger, trustless system, and immutability not only makes it a disruptive innovation in the electronic payment industry but also potential solutions for other areas that require trust establishment. Electronic voting (E-voting) scheme is a use-case where all attributes of blockchain can offer a mechanism for an open, fair and universally verifiable electoral process. In this paper, we review the requirements and then propose Votereum, an E-voting system that utilizes the blockchain technology. The proposed system is empowered by Ethereum platform, including one server manages the entire system and the other handles all blockchain-related requests. The implementation is also deployed to Rinkeby testing network for evaluation on the feasibility and discussion on some security concerns, which are mentioned in the conclusion of this paper.**

Keyword—blockchain, E-voting, Ethereum, smart contract

I. INTRODUCTION

Election is a formal process that allows members of a political body to vote and select individuals to become their leader. In this way, it has great power that impacts the fate of a nation [1]. This is a widely used method all over the world and is a symbol for modern democracy [2]. Thus, this process needs to be secured, fair and provides an easy, convenient way for its citizens to access.

There are many forms of electoral procedures, varying from the traditional paper ballot to E-voting. The traditional way of using paper ballot is the most popular and easiest to access, however, it still exists a critical issue. According to the founder of Follow My Vote – Nathan Hourt, the integrity of the paper-based election still relies on the trust that the officials conducting their jobs correctly and honestly throughout the process. Therefore, there are chance that the result may have tampered without people's awareness.

Despite what voting scheme is used, the final and crucial objective is to ensure the security, transparency, preventing the result from manipulation while preserve voter's anonymity. E-voting has long been an interest to researchers for the past decades [3]. They are devoting to build an efficient system that reduces the cost of organization and solves the current limitations while satisfies the expected requirements. Nowadays, election machines have been used in a few countries, however, they are flawed due to insufficient security measures and physical vulnerability. These machines can be sabotaged under physical force which affects the ballots in the same way of traditional voting.

Blockchain technology is one of the promising solutions for those issues. Blockchain is a network consisting of many peer-to-peer nodes connected and synchronously maintaining an exact version of the database, also known as the ledger. Data entry to blockchain must be validated by the major of network. Therefore, information once stored in the blockchain cannot be modified or deleted and is public to the entire network. These attributes operate based on advanced cryptography, supplying a higher level of security comparing to any previously known database [4]. This makes a blockchain-based E-voting system more reliable and acceptable.

The main contribution of this paper to introduce the design and implementation of Votereum – an E-voting system operates on Ethereum platform which aims to minimize the trust needed in a central authority and enhances the fairness in voting process. The proposed voting scheme is deployed to Rinkeby[1] testnet for implementation and analysis. The system consists of a smart contract written in Solidity language, two functional NodeJS servers and an interface developed with Angular framework.

The content of this paper includes six sections. Section I gives an overall introduction of the paper. Section II presents a deep dive into the underlying technology and related knowledge, as well as a quick peek of existing systems. Section III describes an overall design of the system and real-life scenario. In Section IV, the detailed structure and the implementation of the system application are explained. The subsequent Section V evaluates the deployed system against mentioned requirements for an E-voting system and analysis on pros and cons. Along the side, it discusses about some security concerns and future work. Section VI – the last section is the conclusion on the proposed system.

II. BACKGROUND KNOWLEDGE & RELATED WORKS

A. Blockchain technology & Ethereum platform

Blockchain can be perceived as a public, immutable and distributed ledger that stores records of data and is controlled by a large number of interconnected nodes. Data is appended via transactions under the agreement of network's majority. Each of these nodes maintains their own copy of the distributed database containing the history of all transactions that the network has processed. Transaction once stored into the blockchain cannot be altered or deleted. For each transaction successfully confirmed into the network, a hash value is produced which serves as a transaction identification that can be used to search for the transaction information. Transactions are packed into blocks that link together sequentially, in which the next block making process must reference the previous block hash value [1]. Attempts to

[1] Rinkeby is a testing, public blockchain for development of blockchain-based application. More at: https://www.rinkeby.io

978-1-5386-9313-1/19/$31.00 ©2019 IEEE

change is more difficult since it has to change the next block. Changes that are made by a particular node will be different from the database shared by other nodes therefore becomes invalid and will be rejected. Blockchain operates as a trustworthy service maintained by a group of non-trusting parties. Thus, it can be considered as a reliable third-party that mediates changes and provides secure computing machine [5].

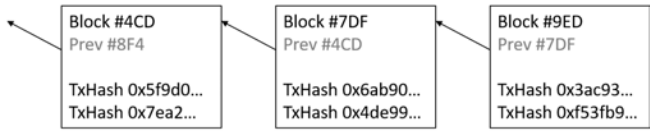

Figure 1. Linked block in blockchain

Ethereum is one of the most popular blockchain platforms nowadays [6]. One important component of Ethereum that contributes to the potentials of its application to real life is Ethereum smart contract. There are two types of account in Ethereum, each account has its own address that identify itself in the network [7]:

- **External Personal Account (EOA):** This is a key pair owned by users, allowing them to create transactions, receive or transfer fund in network.
- **Contract Account:** This is a smart contract address deployed through a transaction by an EOA.

Smart contract can be considered as executable lines of code. This code provides functions to interact with the contract indicates what information in the transaction is checked and what following action will be performed[8]. Smart contract is self-verifying and self-executing, assuring that actions are executed under the party's agreement. This creates a new type of trust connection without relying on a single party [4]. With the help of smart contract, blockchain technology is therefore expected by many to be an ideal tool for election since the voting process flow is conducted automatically and is traceable and verifiable. Votereum is designed to utilize the advantages of blockchain with smart contract to enhance the election procedure.

B. Properties of an electionic voting system

Ballot privacy and data security during voting process is one of the priorities when designing a voting system. In the last 30 years, the number of researches and studies about protocols used in election is increasing The following are some of the properties proposed in the paper of Cranor [9], Cetinkaya [10] and Fujioka [11]:

Primary properties:

- **Ballot privacy:** The content of the ballot cannot be known to others. No one knows whom the voter voted for.
- **Individual verifiability:** A voter can verify that his/her voted is correctly counted.
- **Eligibility:** Only legitimate voters can participate in the current election.
- **Fairness:** Election outcome is not influenced or manipulated by anyone. The system should not leak early result that affects remaining voters.
- **Uniqueness:** A voter can only cast his/her ballot once. Multiple vote casted is considered invalid.
- **Robustness:** No one can modify or manipulate the voting result as well as when tallying.

Advanced properties:

- **Universal Verifiability:** Anyone can verify the clarity and impartiality of the election.
- **Receipt-freeness:** No receipt can be obtained proving how a voter voted.
- **Coercion-Resistance:** There is no cooperation between a voter and a coercer to prove how a vote was casted.

Depends on the electoral procedure and selected voting schema, there are trade-offs among the above properties or among those properties and scalability, security and system's availability. The selection of suitable properties should rest with the scale and demand of the election.

C. Related blockchain-based E-voting system

With the birth of blockchain technology, modern e-voting system designs has utilized the existed researches and combined with advanced properties of new technology to raise people expectation about an optimal solution for those problems. Two blockchain-based voting system that this paper has surveyed were BlockVote by Yifang Wu [2] and EthVote by Matthew Flint [12], which used additional techniques to hides voter's identity while still guarantee a transparent and secured election.

For BlockVote, it was built based on Bitcoin blockchain and Ring Signature scheme. In general, Ring Signature provides a way to verify if an individual belongs to a group without reveal that person's identity [13]. One drawback mentioned in its document is the protocol works efficiently for less than 3000 users due the limitation of Ring Signature algorithm. Additionally, since Bitcoin is invented for the purpose of an online payment system, it's not utilized for application development. Thus, the system servers handle most of the important operations including the tallying phase. This might increase the risk of fraud and severely affects the final result.

EthVote, on the other hand, is a decentralized application that runs on Ethereum blockchain and take advantage of smart contract. In this system, only eligible voters are allowed to vote while their identity remains anonymous. This is achieved by the use of Blind Signature in which a message is signed without its content being revealed and can then be verified just like a regular digital signature [14]. Similar to BlockVote, EthVote is dedicated to creating a secure and transparent E-voting system, while preserve voter privacy. With the use of smart contract, the tallying phase is conducted in the blockchain. This will enhance the fairness as it prevents fraud from occurring and the community can later verify how the ballots are counted.

The design of Votereum is inspired by EthVote but simplified the structure for a quick and easy establishment. Yet, it still provides a scheme that is verifiable and fulfil some of the requirements for an E-voting system. For EthVote, the work of funding voter is carried out at system level. This might contribute to the increased fraud risk as servers are more vulnerable to attacks. Votereum helps solve this problem by letting smart contract do the work. The proposed system also reduces the transaction cost of the deployment of some work comparing to the original system. Furthermore, it improves the user interface that grants easy access to the system.

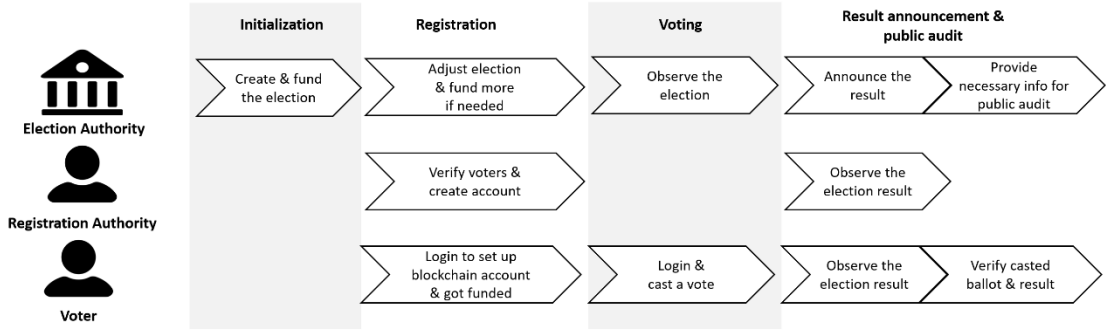

Figure 2. Entities in voting process

III. SYSTEM DESIGN

In this paper, we propose an E-voting system used on a national scale. *Figure 2* gives an overview of voting process and involved entities in a real-life scenario.

A. Entities in system

(i) **Election Authority (EA):** Responsible for setting up and managing the lifecycle of each election. EA has accounts in the system however EA is not eligible to cast a ballot.

(ii) **Registration Authority (RA):** RA verifies voters' identification and register accounts for them in the system. RA accounts are held by officials in polling station located nearby residential areas.

(iii) **Voter:** Citizens are provided with system accounts after authentication of ID cards by RA. Voter can then use their account to log into the system during the election. A blockchain account is required to be set up in order to be eligible to vote. Each account can cast only one ballot and cannot be changed after submitting or send their vote twice.

(iv) **Candidate:** A list of candidates who run for election (including their personal information) is stored in the system database as well as blockchain. They receive votes from voters. Candidates owns no accounts in the system.

(v) **Ballot contract:** Each election has their own smart contract in the blockchain. This contract keeps information about the election itself. It provides functions which allows interaction to the ballot namely start/end phases in voting process, send vote and store the result of the election.

B. Voting schemes

The electoral process consists of five sequential phases as follow.

(i) **Initialization phase:** To begin with, EA should setup a new election, provides necessary information such as ballot name, time for each phase, list of candidates, etc., which deployed a new contract to blockchain. EA will then fund the ballot with an amount of Ether (ETH) [2]. This fund is to later supply the voters, so they can spend for making transaction to the network when voting.

(ii) **Registration phase:** In this phase, citizens bring their passports or ID card to the nearest polling station. RA at the station verifies each person's identity against the citizen database, then generates an account in voting system for him/her. The voters now own accounts in the system and can be participate in the election. The authentication step is to make sure account's information is given to the right one and one can only have one account in the system. Voter who already had a system account doesn't have to register for an account for the next elections.

After receiving the account, voter should log into the system and set up a blockchain account in order to be eligible to vote. This blockchain account however cannot be recover if lost. Upon creation, the account is funded to pay for the fee of making transaction that casts the ballot and add to a list that allows voter to vote in that election's voting phase.

(iii) **Voting phase:** Before starting this phase, the EA is obliged to finalize the ballot. From this point, no more adjustment can be made. In this stage, voters participate in the election by logging into the system using provided accounts and cast their vote. After submitting the vote, a transaction ID is sent back to the voters, which then is used to verify if the vote is recorded into blockchain. Still, voter is not yet able to view or checked the content of their vote. Voters should not reveal who they voted for as well as show the proof of their choice. The election result is either known by anyone.

(iv) **Result announcement and public audit phase:** The result is tallied inside the ballot smart contract and published after voting phase ends. Voters can now check if their vote is correctly counted and recorded into the blockchain. Besides, some necessary information is published for public audit namely ballot smart contract content, lists of blockchain address of whom has voted for each candidate (without disclosing voters identify) and the number of citizens participated in the election. This helps ensure the transparency of the election.

IV. IMPLEMENTATION

This section explains how the entire Votereum system is deployed and running on Blockchain test network – Rinkeby along with the servers using NodeJS and RabbitMQ technology. Angular Framework is used to develop the user interface which intended for all kind of user and easy to use.

A. System architecture

The system architecture is described in *Figure 3* which gives you an overview of the entire system.

[2] Ether (ETH) is the currency of Ethereum system.

- **User interface**: the web interface is developed by Angular. It provides easy access to the voting system.

- **Application server (AS)**: This unit run as an authorization server and a Restful API service. That being said, AS will authenticate access to the API using JWT token. After login to the system, user will receive a JWT token which then will be used to access to the API within their permission level like cast vote, set up a new ballot, etc. For blockchain-related request, AS will communicate with OBR through AMPQ protocol.

- **Online Ballot Regulator (OBR)**: In other to work with Ethereum network, the OBR is responsible for handle request to Ethereum from AS in the request queue. It will execute the functions in the smart contract and wait for the result from the network. Because the response time often takes from three seconds to more than one minute, so the AS and OBR will use AMPQ protocol to send and receive data from blockchain. For example, AS request create a new ballot, it will send to the AMPQ queue and subscribe to this queue, the AMPQ protocol is wrapped up by RabbitMQ – the message broker; it accepts and forwards messages. In other hands, OBR will listen for the message from that queue. After getting the data from Ethereum, OBR will send back it to the AS.

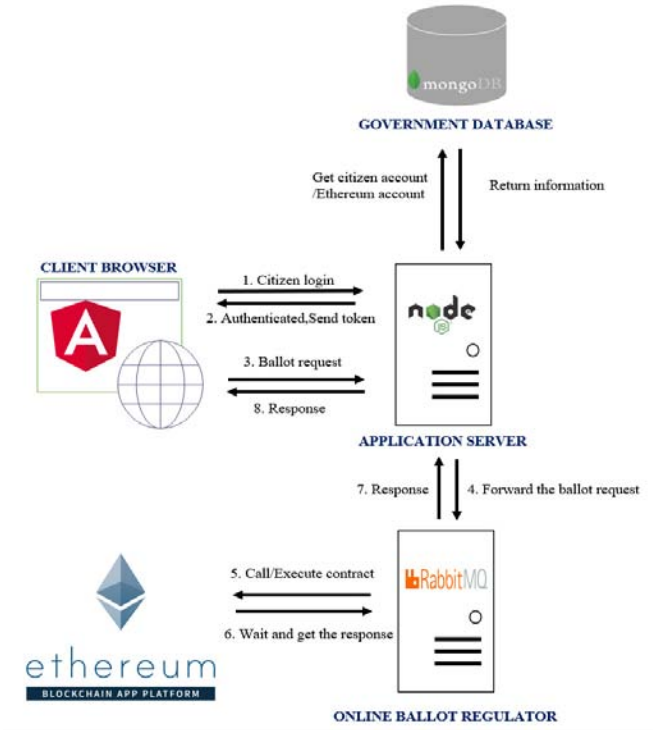

Figure 3. Voting system architecture

- **Local database**: Stores the information of citizen and candidate. The citizen data includes user system account information, Ethereum address and encrypted password (for cast vote), and detail information of that citizen. The candidate data includes their name, citizen ID and some related information about them.

- **Ethereum node**: To interact with the network, an Ethereum node is required to be running in our server which in this case is inside the OBR. This acts as a gateway providing interfaces to interact with the network.

- **Ballot contract:** Each election has their own smart contract and contract address. This contract keeps information about the election itself. It provides functions which allows interaction to the ballot namely to start/end phases in voting process, send vote and store the result of the election.

B. Key functionalities

Before digging into those functions in the smart contract, we should take a look at the variables that store the data of the ballot, citizen, and the owner of the ballot smart contract. The content of the ballot smart contract can be published after the election has ended to provide materials for public audit.

- *owner*: This is the address of the creator of this smart contract, usually it is the address of the EA.

- *ballotName*: Name of the current ballot.

- *startRegPhase*, *endRegPhase*, *startVotingPhase* and *endVotingPhase*: Those are the timeline of the current ballot and will be store as timestamp format. It will begin with the start register date, end register date, start voting date and end voting date.

- *isFinalized*: this variable determines whether the current ballot is finalized. If the ballot is finalized, all the election information cannot be changed.

- *registeredVoterCount*: The number of total voter addresses registered to the current ballot

- *votedVoterCount*: The total number of voters that has cast their vote.

- *storedAmount:* The total fund of current ballot.

- *Voter*: This struct stores four variables that define each voter in this current ballot. Those variables are *eligibleToVote* - A Boolean variable determining whether a voter has rights to cast vote, *isVoted* - A Boolean determine if a voter has voted or not and finally is *votedFor* - this is an array of list candidates that voter cast vote for.

These functions listed below are the key functions that are programmed in the smart contract. It is where the process happens from initialization, administrating who can do what kind of operation, tallying the ballot and when to announce the final result.

- *setupBallot*: This function is only for EA entity to create a new ballot in Ethereum with detail information including the name of this ballot, timeline from registration date to end voting date, the list of candidates, the amount of Ether that will be given to voters and initialize some other related information.

- *onlyOwner (Modifier)*: A modifier is used to restrict functions that only the creator of it can access. It helps limit the number of entities can access important information from those functions.

- *giveRightToVote*: Upon voter has created a blockchain account, this function is called to fund the voter's account and add its address to eligible voter list of that ballot. Voter now is marked eligible to participate in the ballot. This is the improvement of funding operation comparing to EthVote. In EthVote, attacker can interfere with the server and request funding for one account serveral times to accumulate an excess amount of money, which will remain unspent since

user only can vote once. If funding is handled at smart contract, this risk of tampering is minimized.

- *voteForCandidate*: It is used to cast a vote for the selected candidate. Before the candidate gets the vote from a citizen, this function will check for several conditions like if the vote is casted in the voting phase, the citizen has right to vote or he/she has voted before.

- *getCandidateResult:* This function will return the current vote count of each candidate. Anyone can call it when the voting phase has ended.

V. Evaluation and discussion

A. Overall evaluation

The proposed system introduces a concept of a verifiable E-voting platform. Supposing there is no ill intention of interfere in the system from the administrators, the design of this system satisfies some of mentioned properties:

- **Ballot privacy:** As can be seen in Table I, the input data of transaction containing the information of voter's choice is concealed until voting phase ends. Blockchain address does not reveal the identity of its owner since only the network address of voter is shown.

- **Individual verifiability:** An TxHash which is an ID of transaction (as in Table I) is returned to the voter after a vote is submitted. This can be used to verify that the transaction is successfully recorded into the network.

TABLE I
EXAMPLE OF TRANSACTIONS CREATED IN WHEN VOTING

Tx Hash	Block	From	To	Input data
0x9a2a12…	3244868	0x60a939...	0x20cb420…	0xe17b240a000…
0x3d2362…	3240816	0x6123cff…	0x20cb420…	0x431f96a…

- **Eligibility:** Only authenticated citizens that are added into the allowed list of that election can cast their vote.

- **Fairness:** Depends on the election process, the final outcome can be kept until the vote stage ends to avoid impacts on voters' ballots. Another option is to show real time result, this allows candidates to see how effective their electoral campaigns are.

- **Accuracy:** This property is ensured when there is no interference or intended attack between client and server. Result is counted based on data stored in the contract. Tallying method can be checked publicly by auditing ballot smart contract.

- **Uniqueness:** The voter's vote is checked twice, once in the server and once in the contract upon submission to see if they had voted before. The given fund will only suffice for the vote to be caste once.

- **Robustness:** Ballot result and related information is broadcast to the blockchain which is hard to forge and modify.

- **Universal Verifiability:** In the public audit stage, ballot information and any related clues are published. The community can join to verify the result by fetching transactions and do a recalculation.

However, the system does not achieve below properties:

- **Receipt-freeness:** After sending their votes, a transaction ID is returned to the voter as a proof their vote recorded into the blockchain. This however is inescapable as it is how blockchain is built and works.

- **Coercion-Resistance:** There is possibility of vote buying and forced voting when using this system. This is not completely evitable but mitigated by the fact that voter can show the receipt that the vote has been cast but unable to prove who they have voted.

The key advantage of using blockchain-based voting system is to provide a reliable and secured way of election while protect the fairness and transparency of the result. The final outcome is also independently and publicly audited. At the end of the election, fund that remains can be claimed back to the EA blockchain account to avoid fund waste.

When deployed on Rinkeby testnet, error occurs when a larger number of ballots are sent to the network in a short period of time due to the instability of the current blockchain. Thus, the application of this technology to reality needs to be taken in consideration and need more work. It is also important to find a solution for how to deal with the instability of the network and the security during ballot. For extended functions like switching between network can be appended to the system for the flexibility to the cost and various demanded purposes. These can be ideal topics for further study in online voting protocols.

B. Security concerns

1) System vulnerable:

a) DDos: When facing DDoS attack, the Application Server *(in Figure 3)* still depends on the bandwidth and processing ability of the hardware and software. It does not have any ability to identify the DDoS attack early or defend the attack when it happens. This is also a future work to improve system security.

b) Authentication flaw: This is an issue in E-voting system. Even though the authentication process using JWT token [15] to identify citizen after they log in, and the token is valid in a short time, there is a possibility of account being lost or hacked. Currently, our system is working on the web platform, and soon it will be moving to the mobile platform for a more secure and handy for the end user.

2) Privacy rights: Citizens privacy is a large topic in online elections and this is one of the prerequisites. At present, each citizen will associ ate with an Ethereum address. Hence when the election takes place, each vote will be a transaction from an Ethereum account. These accounts will call a function in the smart contract to cast their vote. Because of that, no one will know which citizen's address is, and whom they vote for except that particular citizen.

3) Transparency: With the traditional election, the transparency is not guarantede. Citizens has to trust that their votes are counted and the election council does not manipulate the final result. However, with blockchain technology, the transparency of the election is obvious on the network because it is the nature of its. Every data in blockchain is public and is immutable. Citizens can easily check whether their votes are correctly counted by using the receipt of their vote. The result and electoral procedure can also be verified by the community.

TABLE II
GAS COSTS COLLECTED FROM THE CONDUCTED EXPERIMENTS ON THE RINKEBY TESTNET

Ballot type	**Gas cost (unit)**			
Timeline	Create ballot	Register	Vote	Total
3 Candidates and 3 voters	261340	424536	380176	1066052
3 Candidates and 5 voters	261340	677560	805127	1744027
5 Candidates and 3 voters	261532	424536	542630	1228698
5 Candidates and 5 voters	261532	677560	1019248	1958340
6 Candidates and 3 voters	318069	424536	832066	1600770
6 Candidates and 5 voters	318069	677560	1229571	2251107
Initial contract to the blockchain	2240889			

4) The cost: Each transaction has a fixed cost depending on the amount of computational steps it requires, which is measured by a unit called "Gas". It then multiplies with "Gas price" (a price for each gas unit, for this system is the Ether price) resulting in the total cost. Table II gives an overall about the cost of the system in varying types of ballots. Comparing to EthVote, it costs roughly 0.03318 ETH for maximum of 6 candidates in setting up the ballot while Votereum cost around 0.00524 ETH, which is more cost-efficient. The gas cost increases because of the number of candidates and voters involved. In the future, there should be strategy to minimize the cost as the number of candidates and voters increases.

If a public blockchain is used like Ethereum, the cost will fluctuate over the network's current traded price. Building own private blockchain or using another platform should be taken into consideration for cost optimization.

C. Scalability

For Votereum, the use of Ethereum platform is an advantage for the fast deployment of the system, but the scalability of the system is limited with the growth of the network. This is because of a large number of transactions made in Ethereum, more than 450000 transactions per day *(according to etherscan.io/chart/tx)*. Therefore, we need to study other platform or build up our own blockchain network to scale up the system in the future.

VI. CONCLUSION

Online voting is the future because everything geared toward easy usability for the end user. The objective of building an E-voting scheme on top of blockchain technology is to make the electoral process faster, easier while reduce the cost of holding a traditional election. It also aims to remove the existing security concerns of the current traditional voting scheme. The most crucial issue that may lead to those concerns is about the trust establishment between voter and election authority. By adopting blockchain advantages, the need of trust in individuals can be minimized since blockchain operates under no control of single entity.

In this paper, we proposed Votereum, an Ethereum-based E-voting system that utilizes blockchain technology and smart contract to enable an open, secure election while protecting voter's privacy. We have outlined the requirements, architecture, design of the system and a real-life scenario of mentioned voting scheme. The system has fulfilled the primary requirements of ballot privacy, uniqueness, universal verifiability and robustness. However, it does not satisfy the needs of receipt-freeness and coercion-resistance. By using our voting scheme, voters are allowed to be verified and vote at their nearby resident election station which could potentially increase voter turnout.

The implement of our system successfully deployed to Rinkeby testnet shows the feasibility of building such blockchain-based system that optimizes the electoral process. Despite there is vulnerability in our system's security of single point failure and the possibility of dishonest officials interfering with the system is not completely prevented, it gives us a look how blockchain can change the way we vote and make election better. Votereum can be considered as a prototype for blockchain-based E-voting and it is still far away from a system that can solve real work problem. However, a lot of future work can be done to support the application of blockchain in E-voting to enable a quick, democratic voting way in the future.

ACKNOWLEDGEMENT

This research is funded by University of Information Technology-Vietnam National University HoChiMinh City under grant number D1-2019-01

REFERENCES

[1] R. Hanifatunnisa and B. Rahardjo, "Blockchain based e-voting recording system design," *Proceeding 2017 11th Int. Conf. Telecommun. Syst. Serv. Appl. TSSA 2017*, vol. 2018–Janua, pp. 1–6, 2018.

[2] Y. Wu, "An E-voting System based on Blockchain and Ring Signature (Thesis)," *dgalindo.es*, 2017.

[3] L. Fouard, M. Duclos, and P. Lafourcade, "Survey on electronic voting schemes," *Support. by ANR …*, 2007.

[4] F. Þ. Hjálmarsson, G. K. Hreiðarsson, M. Hamdaqa, and G. Hjálmtýsson, "Blockchain-Based E-Voting System," p. 0, 2018.

[5] C. Cachin and M. Vukoli´cvukoli´c, "Blockchain Consensus Protocols in the Wild," 2017.

[6] Katalyse.io, "Ethereum Platform - What You Should Know – CryptoDigest." [Online]. Available: https://cryptodigestnews.com/ethereum-platform-what-you-should-know-900ff0b5b5ec. [Accessed: 24-Oct-2018].

[7] F. L. Meeser, "Decentralized , Transparent , Trustless Voting on the Ethereum Blockchain," pp. 1–6, 2017.

[8] W. Prinz and A. T. Schulte, "Blockchain and Smart Contracts Technologies , research issues and applications," 2018.

[9] L. F. Cranor and R. K. Cytron, "Sensus: a security-conscious electronic polling system for the Internet," in *System Sciences, 1997, Proceedings of the Thirtieth Hawaii International Conference on*, 1997.

[10] O. Cetinkaya and D. Cetinkaya, "Towards secure E-elections in Turkey: Requirements and principles," in *Proceedings - Second International Conference on Availability, Reliability and Security, ARES 2007*, 2007.

[11] A. Fujioka, T. Okamoto, and K. Ohta, "A Practical Secret Voting Scheme for Large Scale Elections," in *Proc. of Advances in Cryptology*, 1993.

[12] M. Flint, "The Future of Democracy: Blockchain Voting," no. April, pp. 1–21, 2016.

[13] R. L. Rivest, A. Shamir, and Y. Tauman, "How to Leak a Secret: Theory and Applications of Ring Signatures," in *Theoretical Computer Science - Essays in Memory of Shimon Even*, 2006.

[14] "Blind signatures." [Online]. Available: https://www.cs.bham.ac.uk/~mdr/teaching/modules06/netsec/lectures/blind_sigs.html. [Accessed: 31-Dec-2018].

[15] "JSON Web Token Introduction - jwt.io." [Online]. Available: https://jwt.io/introduction/. [Accessed: 07-Jan-2019].