National College of Ireland

# An Evaluation of Privacy Enhancing Technologies for Blockchain Based Voting

MSc Research Project
MSc Cybersecurity

Noel Varghese Oommen
Student ID: 23210567

School of Computing
National College of Ireland

Supervisor: Joel Aleburu

**National College of Ireland**

**MSc Project Submission Sheet**

**School of Computing**

**Student Name:** Noel Varghese Oommen

**Student ID:** 23210567

**Programme:** MSc Cybersecurity **Year:** 2025

**Module:** Practicum Part 2

**Supervisor:** Joel Aleburu

**Submission Due Date:** 29/01/2025

**Project Title:** An Evaluation of Privacy Enhancing Technologies for Blockchain Based Voting

**Word Count:** 6119 **Page Count:** 18

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

**Signature:** *Noel*

**Date:** 29/01/2025

**PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST**

| | |
|---|---|
| Attach a completed copy of this sheet to each project (including multiple copies) | □ |
| **Attach a Moodle submission receipt of the online project submission,** to each project (including multiple copies). | □ |
| **You must ensure that you retain a HARD COPY of the project**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. | □ |

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

| **Office Use Only** | |
|---|---|
| Signature: | |
| Date: | |
| Penalty Applied (if applicable): | |

# An Evaluation of Privacy Enhancing Technologies in Blockchain Based Voting

Noel Varghese Oommen
23210567

**Abstract**

This paper examines how privacy-enhancing technologies such as Zero Knowledge Proofs and Self Sovereign Identity help in improving the privacy and compliance challenges faced by blockchain. For this, a decentralized voting system was taken as a benchmark for the evaluation. The evaluation focuses on the performance, privacy, and compliance factors of implementing each of the privacy-enhancing technologies to understand the scope of scalability and practicality of integrating them. The implementation uses Circom to create the zero-knowledge proofs and Privado ID to test self-sovereign identity. From the analysis, nine different compliance challenges were identified, of which eight of them were addressed by the proposed design. It was also discovered that it is more feasible to deploy the platform in a Layer 2 blockchain such as Polygon compared to a Layer 1 blockchain such as Ethereum due to the lower transaction latency and transaction cost the latter provides.

# 1 Introduction

Privacy has become the need of the present times, and so is the need for a secure and transparent voting system that can ensure democratic integrity. There is a growing distrust in the conventional voting systems. The 2020 US election, for example, highlighted the need to have transparent and immutable voting systems. In a poll conducted by Ipsos, a staggering 30% of the registered voters believed that the election results were rigged (Jackson, Lohr and Rollason, 2024). This is not an isolated incident, as studies show that there has been a growing distrust in the electoral process since the early 2000's (Sances, 2023). The 2016 U.S. presidential election showed concerns regarding Russian interference in the election (Blake.J, 2020). With features such as immutability, decentralization, cryptographic security, and transparency, blockchain has emerged as a sensible solution to address the challenges of current voting systems. But the same transparency and immutability of blockchain raises privacy concerns, that can expose the identity and choice of the voters publicly. This poses a serious violation of data protection laws such as GDPR and election security frameworks by organizations such as NIST. A detailed list of the specific compliance conflicts and their possible solutions are listed in the sections below.

**The aim of this research** is to investigate the feasibility of using privacy-enhancing technologies that can be used to make blockchain-based voting systems more compliant with regulations and how effective they are in addressing the identified challenges.

**The Research Question** the above research problem motivates is: What is the feasibility of implementing privacy-enhancing technologies in blockchain voting and how does it improve privacy and compliance?
**The major contribution** of this research is the evaluation of the feasibility and privacy benefits of implementing zero-knowledge proof and self-sovereign identity in blockchain voting and its effectiveness in addressing the compliance challenges.

The structure of the document is further divided into six main sections, which include the discussion of relevant literature in the related work section, the research methodology in section 3, a design specification in section 4, the implementation detailed in section 5, its evaluation in section 6, discussion of the finding in section 7 and finally the conclusion and future works in the 8th section.

# 2 Related Work

## 2.1 Compliance and privacy challenges in blockchain

Initially, when conceptualized, Nakamoto in his white paper for Bitcoin argued that the privacy provided by the pseudo-anonymity of transactions in the Bitcoin blockchain network was sufficient. This is because, even though transactions are public since it doesn't link to any personally identifiable details, it was considered to be private enough and untraceable. But with the modern clustering and analysis tools we have today, this pseudo-anonymity has become obsolete (Vitalik Buterin et al., 2023). Tools like BACH can use multiple heuristics simultaneously to detect patterns in transactions, cluster addresses, and map transaction flows (Caringella et al., 2024). However, these clustering algorithms still have scalability issues and false positives, but with the rapid growth in the field, better algorithms, and analysis tools are coming up that can give more accurate results, which pose a great threat to pseudo-anonymity of addresses in the blockchain network. In a study conducted on the blockchain Monero by Xiaoqi Li, they explain the transaction privacy leakage in Monero and show how they can predict the real transaction amount with an accuracy of 80% even while using chaff coins to hide the actual amount (Li et al., 2017). According to a study conducted by Rahime, the main areas of conflict are the right to erasure, right to rectification, data minimization, purpose limitation, and storage limitation. Personal data and data subjects are the core of GDPR. The immutable nature of blockchain makes their coexistence with data protection laws such as GDPR challenging. Personal data and data subjects are the core of GDPR (Belen-Saglam et al., 2023). Apart from GDPR, there are other governmental bodies that have made standards for electoral privacy. The Cybersecurity Framework Election Infrastructure Profile by The National Institute of Standards and Technology (NIST) highlights the need for ensuring the privacy and security of the voter registration data (Brady et al., 2024). On-chain registration lacks privacy and off-chain registration conflicts with the decentralized nature of blockchain, thus we need an effective way of user registration. One such method is the use of blockchain integratable self-sovereign identity for voter KYC.

## 2.2 Blockchain and voting: A general overview

Decentralized voting was one of the first applications of blockchain. The immutable, transparent and decentralized nature of blockchain is highly attractive for building a voting system everyone can trust (Jun Huang., 2022). There has been many attempts to materialize this concept in real elections, but this is still far from becoming a reality due to privacy and scalability issues. There have been many attempts to bridge this gap, one such attempt is the use of Two-Factor Authentication (2FA) for enhancing security (Abayomi-Zannu et al., 2020). Even though this method does help in improving the security of the voting system, since 2FA requires linking the voter's identity to the system, it defeats one of the key features of decentralized voting, i.e. anonymity. Another paper suggests off-chain KYC for registering each voter by submitting government issued identity documents to be verified by a certification authority (S. Venkatramulu et al., 2024). But this can severely affect the scalability of the system, the anonymity of the voter and conflicts with the decentralized nature of blockchain. This is where the Self Sovereign Identity can help in fixing the security and verification challenges of decentralized voting, while still maintaining the anonymity and compliance requirements. There are some papers that show the implementation of the voting system in a layer-1 blockchain such as Ethereum, but Ethereum can only handle 15 transactions per second (Antonio de Castro and Coutinho, 2023). Which is not practical in a real large-scale voting scenario. Hence it is better to implement the voting system in a layer-2 blockchain such as Polygon that can handle thousands of transactions per second for a fraction of the transaction fee. Private blockchains are another way to improve the privacy aspect of the voting system, in a paper by Shantanu, they implemented a voting system on Hyperledger Fabric which is an open sourced permissioned blockchain framework. From their implementation it is clear that the permissioned nature of the blockchain and the strict access control makes it much more compliant than public blockchains (Vidwans et al., 2022). But since a permissioned blockchain is controlled by a predefined group of nodes, there exists a risk of insider threat, where a set of nodes can collude to influence the outcomes of the election. Hence a public blockchain is the best option as it is hard to pull of an insider threat to the public nature of the blockchain.

## 2.3 Privacy and performance enhancing technologies in blockchain

The privacy issues in blockchain calls for privacy enhancing technologies. There are many such technologies that are being integrated into blockchain in recent years. Homomorphic encryption is one such technology in which we can perform computation on encrypted data without the need of decrypting it. In a paper by P Ramesh, they implemented it using the Paillier cryptosystem algorithm to encrypt the voter data and votes (Naidu et al., 2022). This system checks all the boxes in the privacy requirements but the computation of the encrypted data on chain is very resource intensive, which significantly impacts the scalability of the platform for large scale elections. Another such technology is zero knowledge proof, in which we can use a proof to verify whether a statement is true or false without revealing any information. In a paper by Yuxiao Wu, they show the implementation of zero knowledge proof for a blockchain based voting system (Wu and Kasahara, 2023). But in their system the voter registration and authentication are done off-chain by humans which makes the process much slower and can results in the exposure of the voter's personal data.

## 2.4 Gaps in the literature and Conclusion

From the review of literature, it is clear that there are numerous privacy, performance and compliance challenges in the implementation of a large-scale decentralized voting platform. Combining Zero Knowledge proofs and Self Sovereign Identity and building it on a layer 2 blockchain such as Polygon seems to be the best way to overcome these challenges. But there are not enough studies that highlight the performance and compliance challenges of implementing them. Thus, this paper focuses on evaluating the performance, privacy and compliance aspects of these privacy enhancing technologies.

# 3 Research Methodology

The research methodology consists of three different stages namely developing the code, voting test and evaluation. From the evaluation an inference is made regarding the effectiveness of the systemFigure 1 shows a clear illustration of the research methodology.

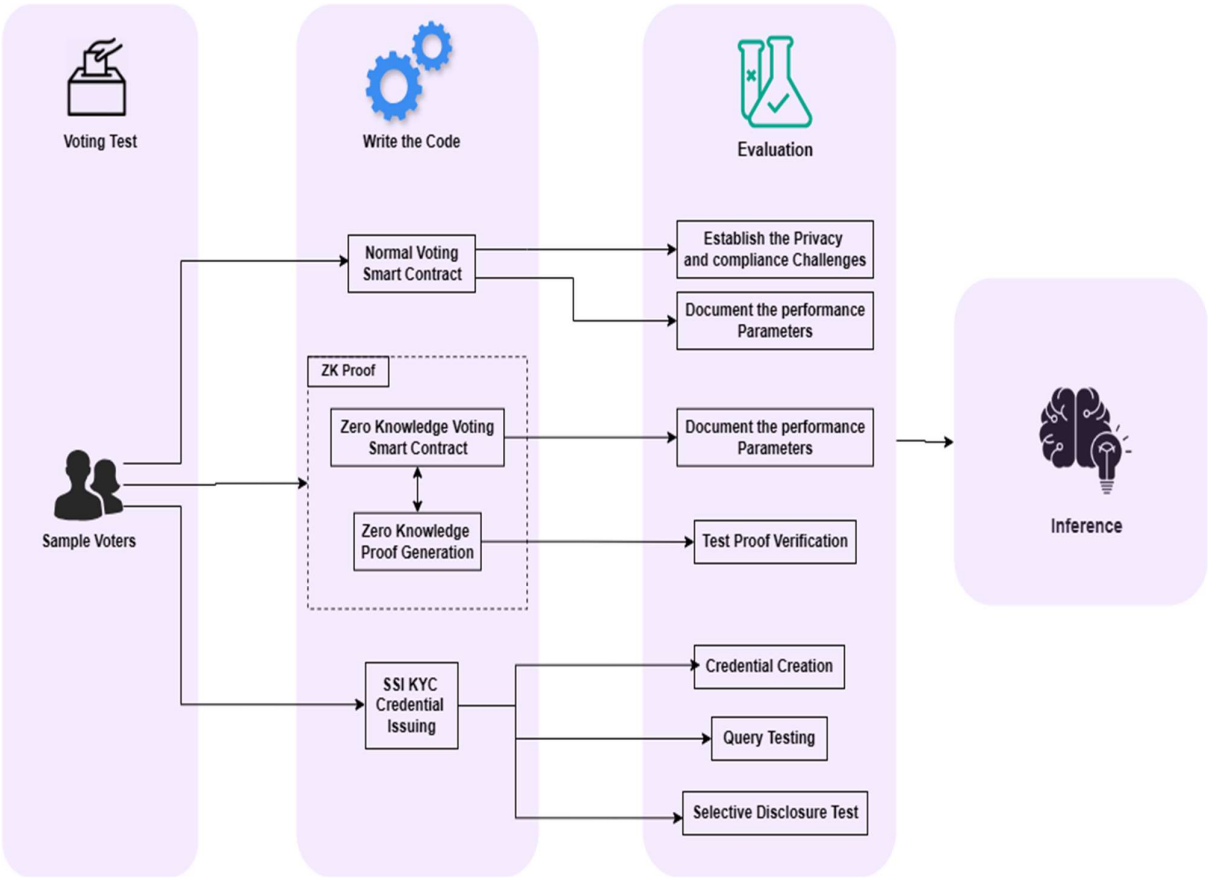


**Figure 1: Research Methodology**

This research follows a mixed method research methodology that combines both quantitative and qualitative analysis. It focuses on evaluating the feasibility of implementing privacy enhancing technologies in blockchain voting. Zero Knowledge proof and Self Sovereign Identity were chosen as the choice of privacy enhancing technologies for the evaluation as both these technologies together can address all the main privacy and compliance challenges that

exist in blockchain voting systems. This is because Zero knowledge proof can mask the choice of candidate thus helping us leverage the transparency of transactions on a public blockchain while still maintaining privacy of the voters. Also, there has been major developments in the Zero knowledge space and its integration with blockchain, with tools such as Circom and Zokrates that have functions that enable easy integration with blockchain. Circom has a function that can automatically generate a solidity code for verifying the zero-knowledge circuit created, which can then be easily deployed to a blockchain. Self-Sovereign Identity helps in solving the KYC issues in blockchain voting by providing a secure and private method for issuing and authenticating credentials. Alternatively other privacy enhancing technologies such as Ring signature, Multiparty Computation (MPC) and Fully Homomorphic Encryption (FHE) were considered. In the case of ring signatures although it provides anonymity it has difficulties in balancing traceability and anonymity, and it can be broken with quantum computing (Perera et al., 2022). FHE is a great option because we can tally the encrypted votes without having to decrypt them, but they are computationally very intensive, which severely impacts their scalability (Ong et al., 2024). Multi Party Computation requires communication between the two parties to complete the computation, which is not practical in this scenario (Liu, 2024).

The voting platform was deployed and tested on two different public blockchain networks, that are, the Sepolia test network of Ethereum which is a layer 1 blockchain, and the Amoy test network of Polygon which is a layer 2 blockchain on Ethereum. Ethereum was an obvious choice due to its rich and mature ecosystem and my personal familiarity with the solidity programming language of Ethereum. Alternatively, the possibility of self-hosting a private blockchain using Hyperledger Fabric was also considered, but it was decided to proceed with a public blockchain due to the global accessibility and the transparency of a public blockchain which would greatly benefit in building public trust on the voting platform (El-Hajj and Bjorn Oude Roelink, 2024). For this setup a Zero knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARK) setup was used instead of Zero-Knowledge Scalable Transparent Argument of Knowledge (zk-STARK). This is because zk-SNARK's have a smaller proof size, takes relatively lesser time for verification and are not as computationally intensive as zk-STARK's, making the on-chain verification of the proofs much faster. Zk-STARK's are more scalable for larger computations, but as the computation required for this voting system is very low, zk-SNARK's are sufficient.

The evaluation focuses on the performance and privacy aspects of the implementation. The performance parameters being assessed here are the cost of the transaction, the latency of the transaction, and the gas fee of the transaction. A gas fee is the transaction fee that is paid by a user to compensate the miners/validators of the blockchain for providing the computational power needed for validating the transaction (Meister and Price, 2024). Each of these tests is to be carried out on both the normal voting platform and the privacy-enhanced voting platform. This methodology ensures a comprehensive assessment of privacy enhancing technologies and their feasibility in a blockchain based voting scenario providing both theoretical and empirical insights.

# 4 Design Specification

For evaluating the efficiency and scope of implementing the privacy enhancing features to the voting platform, three separate tests were created. The first design is for evaluating the normal voting system. The design used here is plain and direct as shown in Figure 2.


**Figure 2: Normal Voting using blockchain.**

In this a solidity contract for voting is compiled using Remix.ide and deployed on to both the Ethereum and Polygon blockchain networks. Remix IDE is an open-sourced web based Integrated development environment (IDE) which is used for writing, testing, debugging and deploying smart contracts to the Ethereum blockchain (Amir Latif et al., 2020). The voting mechanism used here is a direct on-chain voting, where each voter submits their vote directly to the blockchain as a transaction. The transaction outcomes are recorded for analysis. This data is used as a benchmark for comparing the performance using privacy enhancing technologies. Every voter is required to have a Metamask wallet with the test networks Amoy and Sepolia added as a prerequiste with sufficient balance to cast their vote. Once the wallets are setup the voter must connect their MetaMask wallet to the platform to cast their vote. Each wallet address can only cast a single vote, all subsequent attempts to cast a vote by the same address will fail. Every voter address is stored and a “hasvoted” condition is used to check if the address has voted before. This helps in preventing double voting by the same voters. A voter on connecting their wallet address can interact with the platform to cast their vote by typing the candidate's name. If the name is valid, the voter’s MetaMask wallet will be prompted to approve and sign a transaction to cast their vote. On successfully completing the transaction, the vote is cast, and the candidate’s vote count is incremented. If the candidate's name is invalid, the platform will show an invalid candidate message and the transaction will fail. The voting transactions are then viewed through the networks respective block explorer to collect transactions details such as gas fee, time of transaction etc.

The second design is a zero-knowledge proof-based voting system. This design is divided into two parts, an off-chain proof generation using the tool Circom and an on-chain voting Dapp with a verification of the proof as shown in Figure 3.

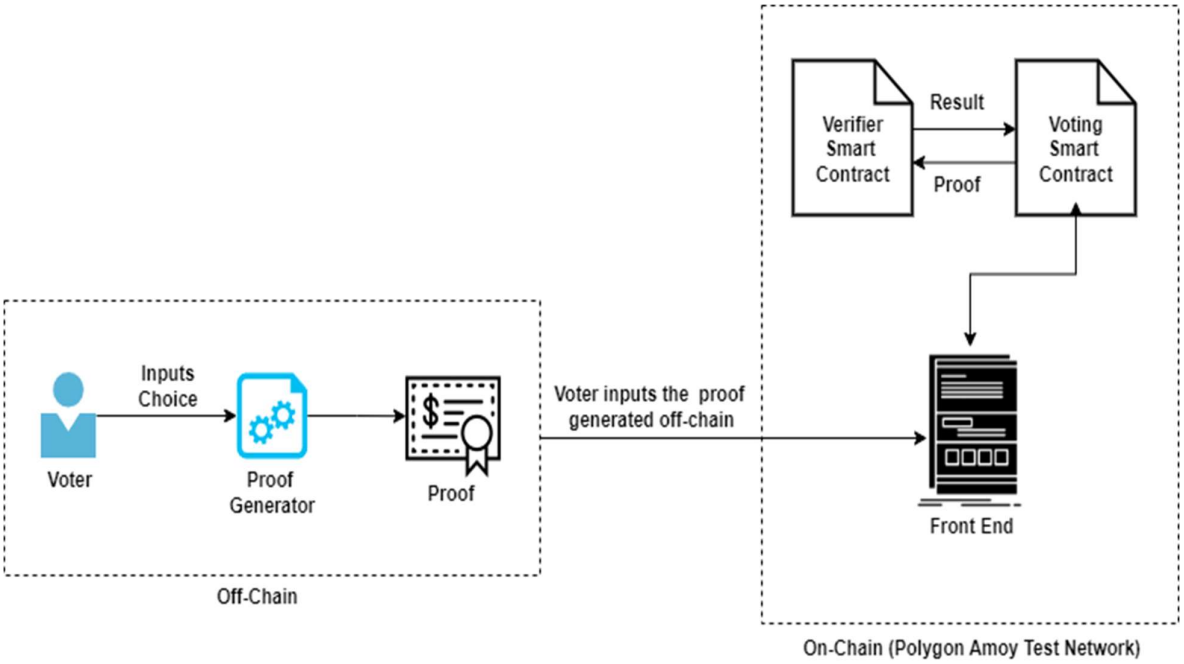


**Figure 3: Zero Knowledge proof enabled voting.**

First the proof is generated off-chain using the Circom proof generator, a detailed explanation of the proof generation is given in the next section. A successful execution of the circom circuit would generate a proof along with both an onchain and off chain verifier, allowing us to verify the proof both in a local computer as well as directly on the blockchain. The front end was automatically generated by the remix ide based on the functions created using the solidity programming language. A user can create a zero-knowledge proof of their choice of candidate using Circom, which would give them a cryptographic sentence. They can then connect to the zero-knowledge voting platform using their MetaMask wallet and then copy and paste the proof generated in the voting text field provided in the front end. This would prompt their MetaMask account to sign and complete a transaction which would register their choice of candidate to the blockchain. This proof is then verified by the verifier using the 'verfier.sol' solidity code or the offchain verification code, which gives a true or false response on accessing the proof submitted. Each choice of candidate will have a separate verifier file. Hence all the proof that gives a true value on running the verification code of the particular candidate will increment the vote count for that candidate. If the proof results in a false value for all the candidate verifiers, it is considered invalid and ignored. This design has the same double voting prevention as the previous design, which allows only a single vote to be cast from a wallet address. Similar to the previous experiment a "hasvoted" condition is used to check if the address has voted before. If the condition fails the transaction fails and the voter is notified.

The third design aims at evaluating self-sovereign identity using PrivadoID. There are two actors in this system, the voter who requests a credential and an issuer who issues and verifies the credentials. The workflow of the design is illustrated in Figure 4.

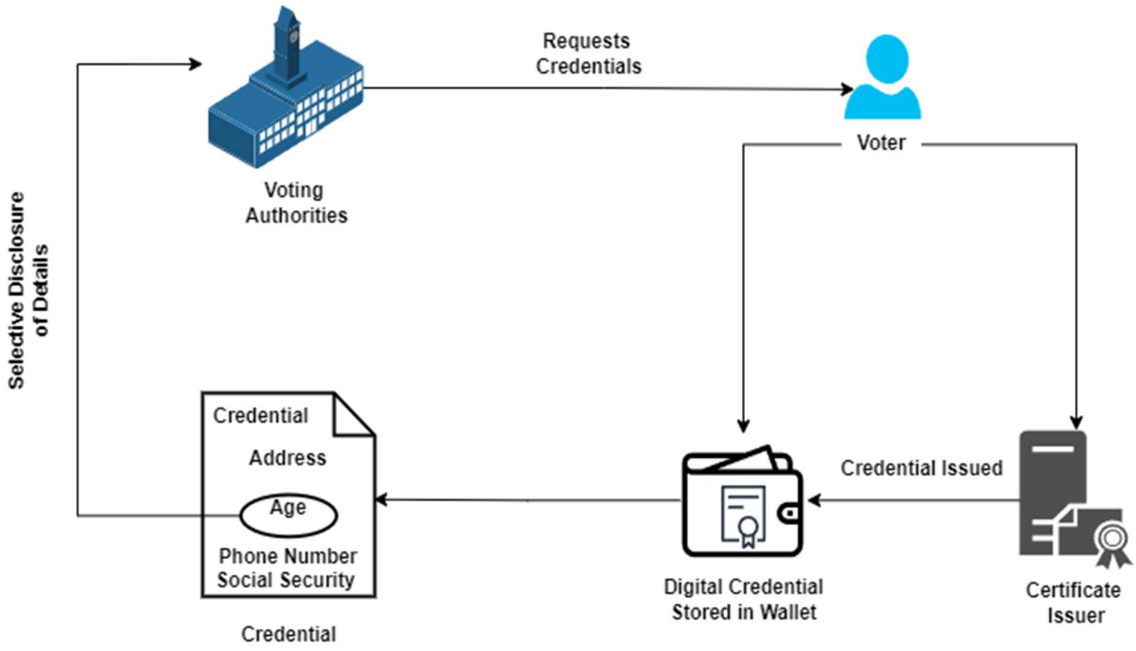


**Figure 4: Self Sovereign Identity design.**

For testing this design, both a voter's perspective and a credential issuers perspective was created and tested. From the voter's perspective, any voter can create a Decentralized Identifier (DID) for free using the PrivadoID website or mobile application. They can make the request for credentials from an authorized issuing body by providing the required information and proof. In this case a KYC credential has to be created for being eligible to cast a vote. On the issuer side the issuer is supposed to create schemas for credential creation and issue the credentials. They are also responsible for creating a query for interacting with the voter. The voter submits the required KYC details selectively without revealing the entire credential. The submitted details are then verified automatically by PrivadoID to confirm the authenticity of the credentials. The result of verification is then displayed on the screen. This test doesn't lead to voting prompt but is an isolated test to demonstrate the working and evaluate the privacy benefits of using self-sovereign identity in the context of blockchain voting.

# 5 Implementation

In the final stage of implementation for both the normal voting and the ZK voting was implemented on Polygon's Amoy test network, which is a layer 2 blockchain and Ethereum's Sepolia test network, which is a layer 1 blockchain. The smart contracts for both the decentralized applications were coded using Solidity, which is a high-level static programming language used for coding in the Ethereum Virtual Machine (Solidity Programming Essentials, 2018). The zero-knowledge proofs were generated using Circom, which is a language for writing and compiling arithmetic circuits (Bellés-Muñoz et al., 2022). The Zero knowledge proof creation using Circom was done offchain on a local computer. A virtual machine with Ubuntu 24.04.1 LTS was set up with 11 GB RAM, Intel Core i7-7500U CPU 2.7Ghz, 2 cores. In the ubuntu system Circom, Snarkjs and nodejs are installed to create the zero knowledge

circuits and generate the proofs. The proof generation and verification are a seven-step process as shown in Figure 5.

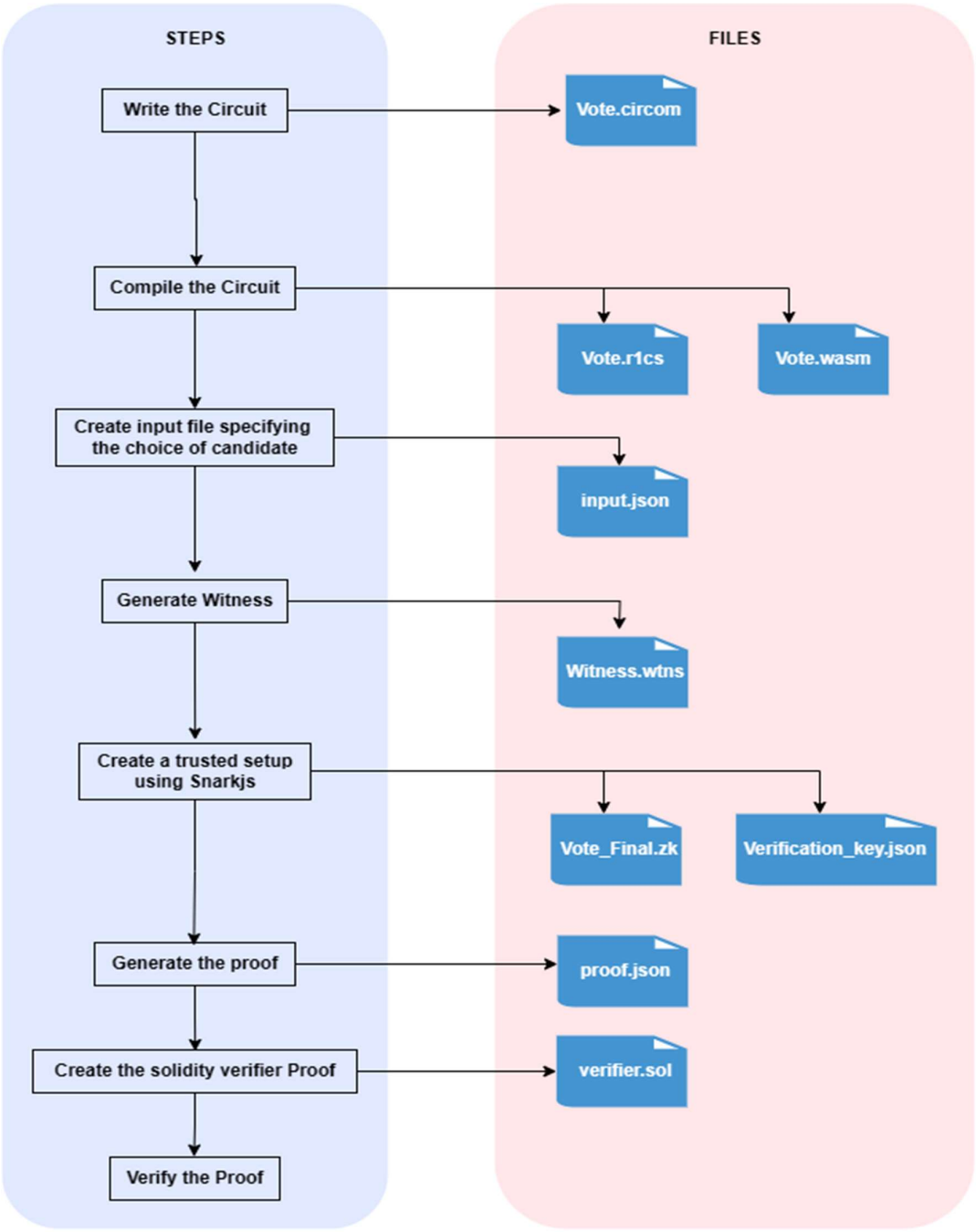


**Figure 5: step by step process of zk proof creation.**

The first step is to describe the constraints of the voting problem using the Circom language to create a circuit called 'Vote.circom'. Once the circuit is created it is then compiled and to generate a 'Vote.r1cs' file and a 'Vote.wasm' file. Rank-1 Constraint System (R1CS) is a low level representation of the constraints of the circuit created, which is used for generating the

proof. The WebAssembly file (WASM) is a compiled logic of the circuit which is used for generating the witness later. An input.json file is created which is used for specifying the choice of candidate for the election. This 'input.json' file together with the 'Vote.wasm' file is used for generating the witness. The witness file is a binary file that satisfies constraints in the circuit with the choice of candidate specified in the input file. Then a trusted setup is generated using Snarkjs to create the public key called 'Verification_key.json' file, which can be used by anyone for creating and verifying the proof. The same step creates a 'vote_final.zkey' which is the final proving key after the randomness was created, it is used for creating the proof. Finally using the witness file and the 'vote_final.zkey' the proof is created as a proof.json file. This proof can be verified using the verification key created earlier in the trusted setup. Additionally, a 'verifier.sol' file is generated, which is a solidity code which can be used for onchain verification of the proof in the Ethereum blockchain (Bellés-Muñoz et al., 2022).
Sample voter DID's were created to simulate the working of the credential issuing and usage. The issuer side was tested by running the issuer node locally on my system on a docker container. For issuing a credential first we must create a schema for the credential using a schema building tool in the issuer node.

# 6 Evaluation

The evaluation focuses on analyzing the privacy, compliance, and performance outcomes of the three experimental setups, that are, a regular voting DApp without any privacy enhancing features, a Zero-Knowledge Proof enabled voting DApp, and a Self-Sovereign Identity (SSI) framework. By comparing the results, the implications of implementing privacy-enhancing technologies (PETs) in blockchain-based voting systems are critically examined. The evaluation combines quantitative metrics, such as gas costs and transaction latency, with qualitative assessments of privacy and compliance. The evaluation mainly focuses on comparing the latency of transaction and the gas fees required. Transaction latency is calculated using the following formula.

$$\textit{Transaction Latency} = T_{conf} - T_{init}$$

Where $T_{init}$ is the timestamp when the transaction was initiated (sent) and $T_{conf}$ is the timestamp when the transaction was confirmed in a block. All latency time is measured in seconds.

## 6.1 Voting without privacy enhancing solutions

The first analysis was done on the normal voting platform with no privacy enhancing features. For this multiple MetaMask wallet addresses were created to simulate the voting process as each candidate can only cast one vote. The first test was carried out in the Polygon Amoy test network, where each voter cast their vote. The voting process was very smooth and the latency and transaction fee analysis of the votes for are given in Table 1.

**Table 1: Analysis of voting in the Polygon Amoy test network**

| Vote | Transaction Fee | Fee in Euros | Gas Fee | Latency |
|---|---|---|---|---|
| 1 | 0.00199358 | 0.001 | 0.000000028254 POL | 4 |
| 2 | 0.00285557 | 0.001 | 0.000000053359 POL | 3 |
| 3 | 0.00246055 | 0.001 | 0.000000035197 POL | 3 |
| 4 | 0.00332259 | 0.002 | 0.000000047045 POL | 3 |

But if we scan the voting contract through Etherscan, we can see all the voters who have cast their votes in the transaction history. On further scanning we can see details about each voter, such as who they voted for and the time of casting their vote. We can also see other transactions made by the voter, such as their digital assets, financial dealings, purchases and all other Web3 activities. This is a serious conflict to compliance regulations such as right to be forgotten, purpose limitation and data subject rights. Using modern clustering software's, we can create behavioral graphs of individuals and understand their lives.

The second analysis was done on a zero-knowledge proof-based voting platform. Exactly like the previous test the votes were cast using the sample accounts, and the results were documented as shown in Table 2.

**Table 2: Analysis of voting in the Ethereum Sepolia test network**

| Vote | Transaction Fee | Fee in Euros | Gas Fee | Latency |
|---|---|---|---|---|
| 1 | 0.0001049955 ETH | 0.374 | 0.00000000150 ETH | 13 |
| 2 | 0.0002470213 ETH | 0.88 | 0.00000000349 ETH | 11 |
| 3 | 0.0001760172 ETH | 0.627 | 0.00000000328 ETH | 10 |
| 4 | 0.0002460126 ETH | 0.877 | 0.00000000351 ETH | 12 |

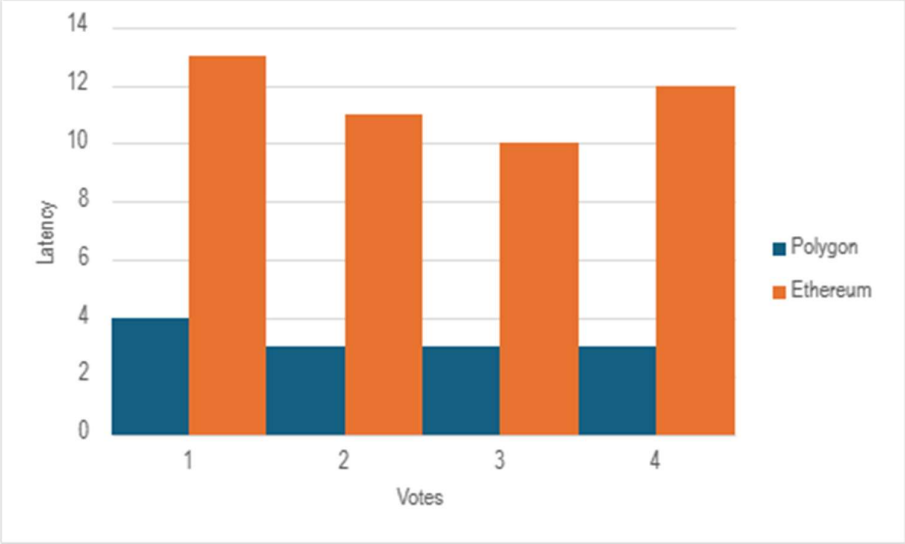


**Figure 6: Latency comparison for regular voting**

The specific compliance challenges that are faced by blockchain voting are listed in Table 3.

**Table 3: Identified Compliance Challenges**

| Compliance Challenges | Explanation |
|---|---|
| Data Minimization | Only collect and process minimal amount of personal data (Voigt and von dem Bussche, 2017). |
| Right to Erasure | Individuals must be able to request the deletion of their personal data (Voigt and von dem Bussche, 2017). |
| Purpose Limitation | Data can only be used for its specified purpose. |
| Data Anonymization and Pseudonymization | Personal data must be effectively anonymized or pseudoanonamized to prevent re-identification. |
| Data Controller and Processor Roles | Clearly define who is responsible for data protection within the system (Voigt and von dem Bussche, 2017). |
| Consent Management | GDPR requires explicit, informed, and revocable consent for processing personal data. |
| KYC | There must be secure identification and verifications procedures in place. |
| Security of Processing | Protect data with strong encryption and security measures. |
| Transparency and Communication | Requires clear communication to voters regarding the handling of their data (Voigt and von dem Bussche, 2017). |

## 6.2 Voting with zero knowledge proof

In the second stage a Zero knowledge enabled voting was carried out. The first part of the process is the off-chain proof generation using Circom, which produced an encrypted proof of the choice of candidate. The proof is then pasted and submitted on the voting platform casting the vote and for on-chain proof verification. The evaluation of the tests is documented in Table 3.

**Table 4: Analysis ZK Voting in Polygon Amoy test network**

| Vote | Transaction Fee | In Euros | Gas Fee | Latency | Proof Generation | Total Time |
|---|---|---|---|---|---|---|
| 1 | 0.8435468428 POL | 0.57 | 0.0000000291 POL | 5 | 10 | 15 |
| 2 | 0.8485101384 POL | 0.57 | 0.0000000291 POL | 7 | 10 | 17 |
| 3 | 0.8476815454 POL | 0.57 | 0.0000000291 POL | 6 | 10 | 16 |
| 4 | 0.8464856472 POL | 0.57 | 0.0000000291 POL | 7 | 10 | 17 |

The average transaction fee is 0.8465560434 POL, the average gas fee is 0.0000000291 POL, and the average transaction latency is 6.25 seconds. This shows that cost and speed of transaction is good enough for real world application. The same test is carried out on Ethereum's Sepolia test network, the results of which are documented in Table 4.

**Table 5: Analysis of ZK Voting in Ethereum Sepolia test network**

| Vote | Transaction Fee | In Euros | Gas Fee | Latency | Proof Generation | Total Time |
|---|---|---|---|---|---|---|
| 1 | 0.10785617457 ETH | 384.622 | 0.00000000368 ETH | 30 | 10 | 40 |
| 2 | 0.10765473214 ETH | 383.902 | 0.00000000366 ETH | 32 | 10 | 42 |
| 3 | 0.10734659731 ETH | 382.804 | 0.00000000363 ETH | 29 | 10 | 39 |
| 4 | 0.10773465913 ETH | 384.188 | 0.00000000367 ETH | 31 | 10 | 41 |

Here the average transaction fee is 0.10764804079 ETH, the average gas fee is 0.00000000366 ETH and the average latency of transaction is 30.5 seconds. This shows a very significant difference in the cost and time of transaction in Ethereum compared to Polygon. The test net tokens don't have a real monetary value, the price comparison given is made based on the value of real Ethereum tokens, to give a common unit for comparison.

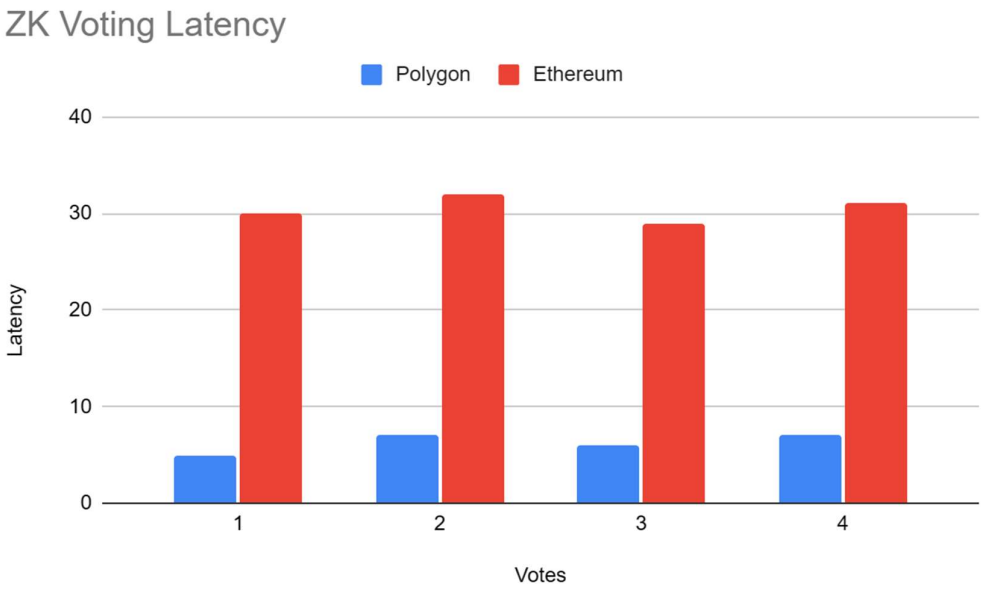


**Figure 7: Latency comparison for ZK Proof enabled voting.**

The implementation of ZKProofs helps in addressing six out of the nine compliance challenges identified in the first experiment. The list of compliance challenges addressed are listed and explained on the Table 6.

**Table 6: Compliance challenges addressed by ZKProofs**

| Compliance Challenges | Explanation |
|---|---|
| Data Minimization | No personal data is collected |
| Right to Erasure | No need of erasure as no personal data is used |

| | |
|---|---|
| Purpose Limitation | Data can only be used for its specified purpose. The proof cannot be decrypted and thus cannot be used for any other purpose |
| Data Anonymization and Pseudonymization | Personal data is effectively anonymized or pseudonymized to prevent re-identification. |
| Consent Management | Low level consent by signing the transaction |
| Security of Processing | Protect data using zero knowledge proof |

## 6.3 Self-Sovereign Identity

A Decentralized Identifier (DID) was created for each of the sample voting addresses using PrivadoID. All of them interacted with a credential issuer that was created earlier to request the KYC credentials by filling in the required details. The credentials are then issued and stored in their PrivadoID wallet. The Query created using the issuer node was put to test for selective disclosure of the KYC details. The query only requests the voters age and asks for consent to share the information. On sharing the data, a verified screen pops up indicating a successful verification. This shows a clear demonstration of data minimization by sharing only requesting the information necessary for verification. Also, as PrivadoID has built in ZK proof, the KYC details being shared are not in plain text but instead are proofs, thus eliminating any chance of data exposure during KYC. The implementation of SSI helps in addressing eight out of the nine compliance challenges identified in the first experiment. The list of compliance challenges addressed by SSI is given in Table 7.

**Table 7: Compliance challenges addressed by SSI**

| Compliance Challenges | Explanation |
|---|---|
| Data Minimization | Uses selective disclosure to minimize exposure |
| Right to Erasure | Individuals have full control over their credentials |
| Purpose Limitation | Data is shared as zk proofs, and thus cannot be used for other purposes |
| Data Anonymization and Pseudonymization | Data is shared as zk Proofs |
| Data Controller and Processor Roles | The credential issuer is responsible for the secure design of the KYC process. |
| Consent Management | GDPR requires explicit, informed, and revocable consent for processing personal data. |
| KYC | Provides a strong KYC mechanism with valid credentials |
| Security of Processing | Credential data is masked as zk proofs |

## 6.4 Discussion

From the analysis above it is clear that deploying the voting platform on a layer 2 network such as Polygon is very efficient. As per the evaluation conducted above it takes less that 7 seconds for the on-chain transaction for voting and 10 seconds for off chain proof generation. The total time of execution is still sufficiently low, which makes the integration of Zero knowledge proofs practically possible in a real-life voting environment. A transaction cost analysis was also carried out to understand the cost efficiency of the system and to compare them between the Ethereum and Polygon blockchain networks. The results are documented in the Tables 4 and 5, the cost is converted to Euros for the ease of comparison. The results show a sharp difference between the transaction costs of Ethereum and Polygon. But the average cost of transaction in Polygon is 57 Cents, which makes the Zero Knowledge voting economically feasible to implement. Privacy of the transactions were also ensured by using these technologies. On scanning the transactions on a block explorer, only the zk proof of the candidate choice was publicaly visible. As shown in tables 6 and 7 by integrating these privacy enhancing technologies together, most of the compliances challenges are addressed. Thus it is clear from the evaluation that using these technologies we can develop an efficient, privacy preserving and compliant voting platform.

# 7 Conclusion and Future Work

As we discussed in the beginning, there is a growing distrust in the current electoral process across the globe. This thesis has critically evaluated the feasibility of integrating privacy-enhancing technologies, specifically Zero-Knowledge Proofs (ZKPs) and Self-Sovereign Identity (SSI), within blockchain-based voting systems. The study focused on three core aspects: compliance with privacy regulations, transaction performance (measured through gas fees and latency), and overall privacy preservation. These evaluations were conducted on two blockchain platforms: Layer 1 Ethereum and Layer 2 Polygon. The proposed solution here takes an approach of data protection by design to minimize the exposure of personal data. Thus, even though the data is immutable and transparent, as it does not reveal any personal data of the voter, it is compliant with regulations. The design suggested here helps in addressing eight out of nine compliance challenges identified. The ninth challenge of communication can be addressed in the future works by integrating technologies such as Ethereum Push Notification (EPNS) that allows in notifying decentralized wallets. Future work would also focus on integrating both zero knowledge proof and self-sovereign identity into a single platform and to automate the generation of the ZK proof locally. The self-sovereign identity can be linked to the voting platform using the PrivadoID API, and a voter can use the credentials by scanning a QR code with their PrivadoID wallet app or by simply providing their DID. The ZK proof generation can by automated by using WebAssembly to automatically generate the proof of the choice of candidate. Layer 2 chains are still improving in performances, and with future developments the transaction latencies will be even faster making the integration more feasible.

# References


Vitalik Buterin, Illum, J., Nadler, M., Schär, F. and Soleimani, A. (2023). Blockchain Privacy and Regulatory Compliance: Towards a Practical Equilibrium. *Blockchain: Research and Applications*, pp.100176–100176. doi:https://doi.org/10.1016/j.bcra.2023.100176.

Caringella, M., Violante, F., De Lucci, F., Galantucci, S. & Costantini, M. 2024, "BACH: A Tool for Analyzing Blockchain Transactions Using Address Clustering Heuristics", Information, vol. 15, no. 10, pp. 589.

Abayomi-Zannu, T.P., Odun-Ayo, I., Tatama, B.F. and Misra, S. (2020). Implementing a Mobile Voting System Utilizing Blockchain Technology and Two-Factor Authentication in Nigeria. Lecture Notes in Networks and Systems, pp.857–872. doi:https://doi.org/10.1007/978-981-15-3369-3_63.

S. Venkatramulu, Rishitha Reddy Gopu, Naresh Badavath, Shreya Karimilla, Sowmith Reddy Arram and Pannala, K.A. (2024). Crypto Ballot: Safeguarding democracy with Blockchain Voting. 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), [online] pp.1–6. doi:https://doi.org/10.1109/icccnt61001.2024.10724481.

Antonio de Castro and Coutinho, C. (2023). Electronic Voting Through Blockchain: A Survey. doi:https://doi.org/10.1109/hora58378.2023.10156749.

Vidwans, S., Deshpande, A., Thakur, P., Verma, A. and Palwe, S. (2022). Permissioned Blockchain Voting System using Hyperledger Fabric. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICIBT52874.2022.9807702.

Naidu, P.R., Bolla, D.R., Prateek G, Harshini, S.S., Hegde, S.A. and Harsha, S. (2022). E-Voting System Using Blockchain and Homomorphic Encryption. 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon). [online] doi:https://doi.org/10.1109/mysurucon55714.2022.9972661.

Wu, Y. and Kasahara, S. (2023). Smart Contract-Based E-Voting System Using Homomorphic Encryption and Zero-Knowledge Proof. Lecture notes in computer science, pp.67–83. doi:https://doi.org/10.1007/978-3-031-41181-6_4.

Jun Huang et al. (2022) 'The Application of the Blockchain Technology in Voting Systems: A Review', ACM Computing Surveys, 54(3), pp. 1–28. doi:10.1145/3439725.

Li, X., Jiang, P., Chen, T., Luo, X. and Wen, Q. (2017). A Survey on the Security of Blockchain Systems. Future Generation Computer Systems, [online] 107, pp.841–853. doi:https://doi.org/10.1016/j.future.2017.08.020.

Belen-Saglam, R., Altuncu, E., Lu, Y. and Li, S. (2023). A systematic literature review of the tension between the GDPR and public blockchain systems. Blockchain: Research and Applications, 4(2), p.100129. doi:https://doi.org/10.1016/j.bcra.2023.100129.

Brady, M., Howell, G., Franklin, J.M., Sames, C., Schneider, M., Snyder, J. and Weitzel, D. (2024). Cybersecurity Framework Election Infrastructure Profile. doi:https://doi.org/10.6028/nist.vts.200-1.

Solidity Programming Essentials (2018). Solidity Programming Essentials. [online] O'Reilly Online Learning. Available at: https://learning.oreilly.com/library/view/solidity-programming-essentials/9781788831383/?sso_link=yes&sso_link_from=NCIRL [Accessed 5 Dec. 2024].

Bellés-Muñoz, M., Isabel, M., Muñoz-Tapia, J.L., Rubio, A. and Jordi Baylina (2022). Circom: A Circuit Description Language for Building Zero-knowledge Applications. IEEE Transactions on Dependable and Secure Computing, pp.1–18. doi:https://doi.org/10.1109/tdsc.2022.3232813.

Perera, M.N.S., Nakamura, T., Hashimoto, M., Yokoyama, H., Cheng, C.-M. and Sakurai, K. (2022). A Survey on Group Signatures and Ring Signatures: Traceability vs. Anonymity. Cryptography, 6(1), p.3. doi:https://doi.org/10.3390/cryptography6010003.

Ong, K., Chen, B.Y., Chi, P.W. and Wang, C. (2024). A Study of Fully Homomorphic Encryption with Evaluation Control. [online] pp.17–24. doi:https://doi.org/10.1109/asiajcis64263.2024.00014.

Liu, T. (2024). Research on Privacy Techniques Based on Multi-Party Secure Computation. 2024 3rd International Conference on Artificial Intelligence and Autonomous Robot Systems (AIARS), [online] pp.912–917. doi:https://doi.org/10.1109/aiars63200.2024.00171.

Meister, B.K. and Price, H.C.W. (2024). Gas fees on the Ethereum blockchain: from foundations to derivative valuations. Frontiers in Blockchain, 7. doi:https://doi.org/10.3389/fbloc.2024.1462666.

El-Hajj, M. and Bjorn Oude Roelink (2024). Evaluating the Efficiency of zk-SNARK, zk-STARK, and Bulletproof in Real-World Scenarios: A Benchmark Study. Information, 15(8), pp.463–463. doi:https://doi.org/10.3390/info15080463.

Bellés-Muñoz, M., Isabel, M., Muñoz-Tapia, J.L., Rubio, A. and Jordi Baylina (2022). Circom: A Circuit Description Language for Building Zero-knowledge Applications. IEEE Transactions on Dependable and Secure Computing, pp.1–18. doi:https://doi.org/10.1109/tdsc.2022.3232813.

Amir Latif, R.M., Hussain, K., Jhanjhi, N.Z., Nayyar, A. and Rizwan, O. (2020). A remix IDE: smart contract-based framework for the healthcare sector by using Blockchain technology. Multimedia Tools and Applications. doi:https://doi.org/10.1007/s11042-020-10087-1.

Jackson, C., Lohr, A.A. and Rollason, C. (2024). Americans accept the election results even if some are unhappy with the outcome. [online] Ipsos. Available at: https://www.ipsos.com/en-us/americans-accept-election-results-even-if-some-are-unhappy-outcome [Accessed 20 Nov. 2024].

Sances, M.W. (2023). Legitimate questions: Public perceptions of the legitimacy of US presidential election outcomes. Research & Politics, 10(4). doi:https://doi.org/10.1177/20531680231206987.

Blake, J. (2020). RUSSIAN INTERFERENCE IN U.S. ELECTIONS: HOW TO PROTECT CRITICAL ELECTION INFRASTRUCTURE FROM FOREIGN PARTICIPATION. Public Contract Law Journal, 49(4), 709–734. https://www.jstor.org/stable/27010377

Voigt, P. and von dem Bussche, A. (2017). The EU General Data Protection Regulation (GDPR). [online] Cham: Springer International Publishing. doi:https://doi.org/10.1007/978-3-319-57959-7.