International Journal of Computer Networks and Applications (IJCNA)
DOI: 10.22247/ijcna/2025/05 Volume 12, Issue 1, January – February (2025)

RESEARCH ARTICLE

An Application-Layer Framework for Privacy Blockchain Transactions and Smart Contract

Souhail Mssassi
National School of Applied Sciences, Cadi Ayyad University, Morocco.
✉ souhail.mssassi@owasp.org

Anas Abou El Kalam
National School of Applied Sciences, Cadi Ayyad University, Morocco.
a.abouelkalam@uca.ac.ma

Received: 20 November 2024 / Revised: 26 January 2025 / Accepted: 11 February 2025 / Published: 28 February 2025

Abstract – While offering transparency and decentralization, Open blockchain networks inadvertently expose user identities and sensitive transaction details. Existing privacy solutions often focus on simple token transfers (e.g., mixers) but fail to protect more complex operations such as smart contract executions. This paper tackles these challenges by introducing a novel application-layer framework anonymizing token transactions and smart contract calls. Building on the principles of Tornado Cash, the approach pools user transactions off-chain, obscuring the link between senders, recipients, and contract interactions. Zero-knowledge proofs were integrated to ensure verifiability without revealing underlying data, all without altering network or consensus mechanisms. Further, a sustainable incentive model is proposed that compensates relayers and executors for gas fees and computational effort, maintaining economic viability. The results indicate that the framework is scalable and platform-agnostic and significantly improves privacy for decentralized applications, mitigating identity exposure and transaction traceability in modern blockchain ecosystems.

Index Terms – Blockchain Privacy, Zero-Knowledge Proof, Blockchain Security, Smart Contract Privacy, Anonymity, Transaction Mixing, Off-Chain Transaction Pooling, Incentive Mechanisms, Multi-Party Computation.

1. INTRODUCTION

Blockchain technology has emerged as a decentralized data management and transaction processing framework. At its core, a blockchain is a distributed ledger that records transactions across a network of computers to ensure transparency, immutability, and security. [1]. Each block in the chain contains a list of transactions, a timestamp, and a cryptographic hash of the previous block, forming an unalterable sequence of records accessible to all network participants (see Figure 1).

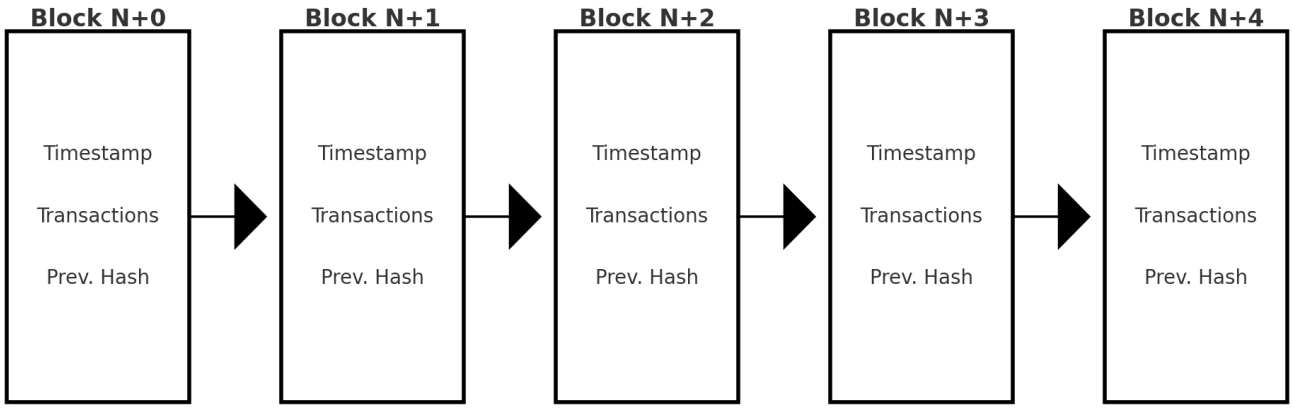

Figure 1 The block structure in the blockchain

The transparency inherent in blockchain systems is a double-edged sword [2]. While it fosters trustlessness, eliminating the need for intermediaries by allowing participants to verify transactions independently, it simultaneously poses significant privacy challenges. Every transaction and smart contract execution is publicly recorded on the blockchain, potentially

RESEARCH ARTICLE

exposing user identities and sensitive operational details. This transparency can lead to the unintended disclosure of proprietary business processes, personal financial information, and other confidential data.

The public nature of blockchain transactions means that, although users' identities are often pseudonymous, sophisticated analysis techniques can de-anonymize participants by linking their addresses to real-world identities. [3]. For instance, patterns in transaction histories can be correlated with off-chain data sources, enabling malicious actors to trace activities back to individual users or organizations. [4]. Smart contracts, which are self-executing contracts with terms directly written into code, exacerbate this issue by making the logic and data of complex agreements publicly accessible.

The lack of privacy in blockchain systems has profound real-world implications:

1. Susceptibility to Targeted Attacks: Exposure of transaction details can make users vulnerable to phishing, blackmail, or other cyber threats. High-value transactions can attract the attention of hackers seeking to exploit security weaknesses [5].
2. Loss of Competitive Advantage: Businesses leveraging blockchain technology may inadvertently reveal strategic information, such as supply chain logistics, pricing models, or contractual agreements, eroding their competitive edge.
3. Regulatory Concerns: Compliance with data protection regulations like the General Data Protection Regulation (GDPR) requires safeguarding personal information [6]. Public blockchains' transparency may conflict with such legal obligations, posing challenges for widespread adoption in regulated industries.

Considering these challenges, enhancing privacy on blockchain platforms is not merely a technical necessity but a fundamental requirement for protecting user interests and fostering broader technology acceptance.

However, most existing privacy solutions largely center on token transfers (e.g., mixers) and do not extend their protective measures to more complex interactions such as smart contract executions. This limitation leaves call parameters, user identities, and sensitive business logic exposed, creating a substantial barrier for organizations seeking robust confidentiality on decentralized platforms

1.1. Problem Statement

Despite decentralized trust's benefits, blockchain environments' inherent openness exposes transactions, contract details, and user behavior to public scrutiny. Participants risk having their identities and proprietary information linked or de-anonymized through sophisticated analysis methods when engaging with smart contracts-beyond simple token transfers. This gap underscores the need for a solution that safeguards financial and non-financial blockchain operations without requiring fundamental changes to existing protocols.

1.2. Research Objectives

1. Enhance Privacy for Smart Contract Calls: Develop an application-layer approach that anonymizes caller identities and contract parameters during execution.
2. Achieve Scalability and Efficiency: Implement an off-chain transaction pooling mechanism to handle a high volume of transactions, reducing on-chain traceability while maintaining acceptable throughput.
3. Introduce Sustainable Incentive Models: Design fair compensation structures tied to transaction complexity and gas fees to encourage the active participation of relayers and executors.
4. Maintain Cross-Platform Compatibility: Operate at the application layer to avoid consensus or protocol-level modifications, enabling flexible deployment across multiple blockchain networks.

1.3. Key Contributions

1. Extension of Transaction Mixing to Smart Contracts: Adapts privacy techniques from token mixers (e.g., Tornado Cash) to protect complex contract interactions, not just financial transfers.
2. Off-Chain Pooling for Enhanced Anonymity: This technique decouples user identities from their transactions by batching them off-chain, making individual transactions indistinguishable on-chain.
3. Zero-Knowledge Verification: This method utilizes zero-knowledge proofs (ZKPs) to validate user actions and ensure transaction integrity without revealing sensitive information.
4. Relayer-Executor Incentive Model: Proposes a fair economic structure for participants who pool and execute private transactions, ensuring both privacy and sustainability of the network.
5. Blockchain-Agnostic Design: Requires no modifications to underlying consensus protocols, facilitating adoption in diverse blockchain ecosystems.

To address these challenges, this research aims to develop a privacy-enhancing flow for smart contract executions that ensures confidentiality and scalability. The proposed solution focuses on anonymizing smart contract calls, protecting caller addresses, and ensuring compatibility with existing blockchain layers. By operating at the application layer, the

RESEARCH ARTICLE

solution avoids modifications to the underlying network or consensus mechanisms, facilitating easier adoption across different platforms. Additionally, a fair and sustainable incentive model is proposed to encourage active participation from relayers and executors who contribute to maintaining privacy.

The paper is organized as follows. Section 1 introduces the privacy challenges inherent in blockchain transactions and smart contract executions, setting the stage for the problem of the research. Section 2 Reviews the existing works in the field, such as Tornado Cash and other privacy-preserving solutions. Their contributions and limitations are analyzed, noting that most existing solutions focus solely on financial transactions and fail to address privacy in smart contract executions, leaving a critical gap in the field. Section 3 Presents the proposed framework, detailing the architecture of the system, the off-chain transaction pooling methodology, and the integration of zero-knowledge proofs (ZKPs) to protect user privacy. This section also introduces the incentive models developed for relayers and executors, which ensure the system's economic viability and network efficiency. Section 4 Evaluates the framework's performance, exploring the relationship between pool size and anonymity, as well as the impact of transaction batching on network throughput and latency. Finally, Section 5 The paper concludes by summarizing the key findings and contributions, emphasizing the scalability, adaptability, and practicality of the privacy-enhancing solution for blockchain-based decentralized applications.

2. RELATED WORKS

Tornado Cash represents a prominent attempt to address privacy concerns within blockchain transactions. It is a decentralized, non-custodial privacy solution on the Ethereum network that enables users to break the on-chain link between source and destination addresses. [7]. By leveraging zero-knowledge proofs (ZKPs) [8], Tornado Cash allows users to deposit Ether (ETH) or other ERC-20 tokens into a smart contract, where the funds are mixed with deposits from other users. The mechanism works as follows:

1. Deposit: A user generates a secret and deposits cryptocurrency into the Tornado Cash smart contract, receiving a corresponding cryptographic note.
2. Mixing: The contract pools these deposits, obscuring the origin of funds by intermixing them with other users' assets (see Figure 2).
3. Withdrawal: The user can withdraw the funds to a new address by presenting the cryptographic note and a zero-knowledge proof that they possess a valid note without revealing their specific deposit.

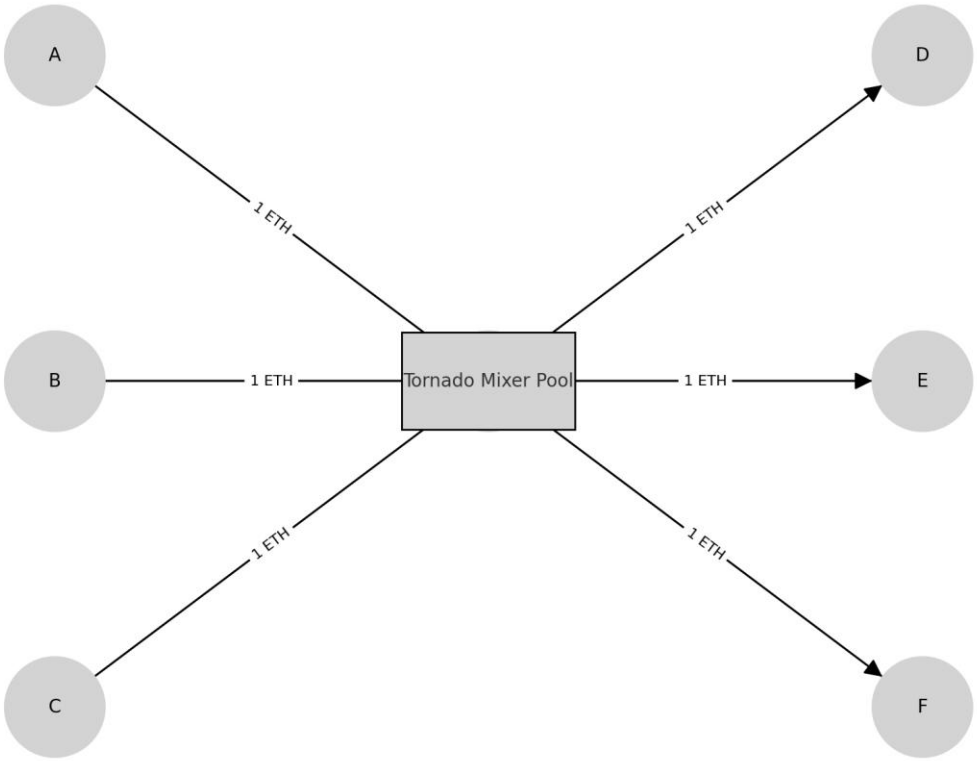

Figure 2 ETH Mixer Pool In Tornado Cash

RESEARCH ARTICLE

This process enhances transactional privacy by making it computationally infeasible to link withdrawals to deposits, thereby protecting user anonymity. While Tornado Cash effectively anonymizes token transfers, it has significant limitations:

- Scope Restriction: The solution is limited to simple financial transactions and only for specific assets, not all assets, and does not extend to more complex interactions, such as smart contract executions, which require additional input data and state changes.
- Post-Withdrawal Privacy: Users must exercise caution after withdrawing funds to avoid patterns that could de-anonymize them [9], such as transferring funds to known addresses or making distinctive transactions.
- Smart Contract Interactions: Tornado Cash does not address privacy concerns when users interact with smart contracts, leaving a gap where sensitive data and user behaviors remain exposed.

From a broader perspective, Tornado Cash's main advantage is its straightforward mixing process and widespread adoption of Ethereum. However, its primary drawback is that it focuses on financial transfers rather than full-fledged smart contract operations. This narrow scope makes it unsuitable for applications that require privacy at the code execution level.

Beyond Tornado Cash, other privacy-enhancing technologies have been developed in the blockchain space. Zcash and Monero are cryptocurrencies specifically designed with privacy features. Zcash uses Zero-Knowledge Succinct, Non-interactive Arguments of Knowledge to allow users to transact anonymously. [10]. Monero employs ring signatures [11], stealth addresses, and confidential transactions to obfuscate transaction details [12].

Zcash offers strong privacy assurances for transactions on its blockchain by leveraging advanced ZKPs (zk-SNARKs). A key benefit is its robust anonymization of senders, recipients, and amounts. However, Zcash's privacy features do not directly extend to smart contract capabilities on platforms like Ethereum. Monero excels in network-level anonymity through ring signatures and stealth addresses, making it one of the more private cryptocurrencies. Despite this, Monero remains a standalone platform primarily focused on monetary transactions, lacking an integrated smart contract environment.

However, these solutions are often limited to native platforms and do not provide privacy features for smart contracts on platforms like Ethereum. Another line of research involves Multi-Party Computation (MPC) protocols, which allow parties to jointly compute a function over their inputs while keeping those inputs private. [13]. While MPC can provide strong privacy guarantees, it often suffers from high communication overhead and complexity, making it less practical for real-world blockchain applications. [14].

For example, MPC-based approaches can theoretically keep contract logic and inputs hidden even from participating nodes. This is advantageous for highly confidential applications. However, the main disadvantage is the heavy computational and communication overhead, which can hinder throughput and make large-scale adoption challenging.

The limitations of these existing solutions can be summarized as follows:

1. Limited to Financial Transactions: Many privacy solutions focus solely on the confidentiality of asset transfers and do not address the privacy of smart contract executions.
2. Compatibility Issues: Some protocols require changes to the underlying blockchain infrastructure or how assets are represented, hindering their adoption across different platforms.
3. Performance Overheads: Cryptographic techniques like zkSNARKs can introduce significant computational and verification overheads, affecting scalability and user experience [15].
4. Trusted Setup Requirements: Protocols requiring a trusted setup [16] Pose security risks, as compromise of the setup phase can undermine the entire system's integrity.

In summary, while several privacy-preserving mechanisms exist—from mixers and specialized privacy coins to MPC-based frameworks—the lack of comprehensive support for smart contract privacy remains a key gap. Moreover, scalability hurdles and the need for trusted setups deter many practical deployments. These gaps underscore the pressing need for an application-layer solution focusing on privacy for contract-level interactions without requiring protocol-level modifications.

Considering these challenges, there is a clear need for a privacy-enhancing framework that supports confidential smart contract executions without imposing substantial overhead or requiring modifications to existing blockchain infrastructures. This research aims to fill this gap by proposing a solution that operates at the application layer, leverages advanced cryptographic techniques, and maintains compatibility with current blockchain ecosystems.

3. METHODS

To address the privacy challenges inherent in smart contract executions on blockchain platforms, a framework is proposed that extends the principles of transaction mixing—traditionally used for simple token transfers—to the domain of smart contract interactions. The proposed approach anonymizes smart contract calls by aggregating transactions from multiple users into a common pool, making individual

RESEARCH ARTICLE

transactions indistinguishable. This aggregation not only obscures the initiators of the transactions but also protects the associated parameters and state changes within the smart contracts.

3.1. Extending Transaction Mixing to Smart Contracts

In traditional transaction mixing, as implemented by solutions like Tornado Cash, users' token transfers are pooled together to break the link between senders and recipients, enhancing transactional privacy. Similarly, the framework collects smart contract transactions from various users into an off-chain pool. By doing so, an anonymity set [17] It is created where each transaction is indistinct from others in the pool (see Figure 3), making it computationally infeasible for an observer to link a specific transaction to a user. This method effectively decouples users' identities from their transactions and the associated smart contract data.

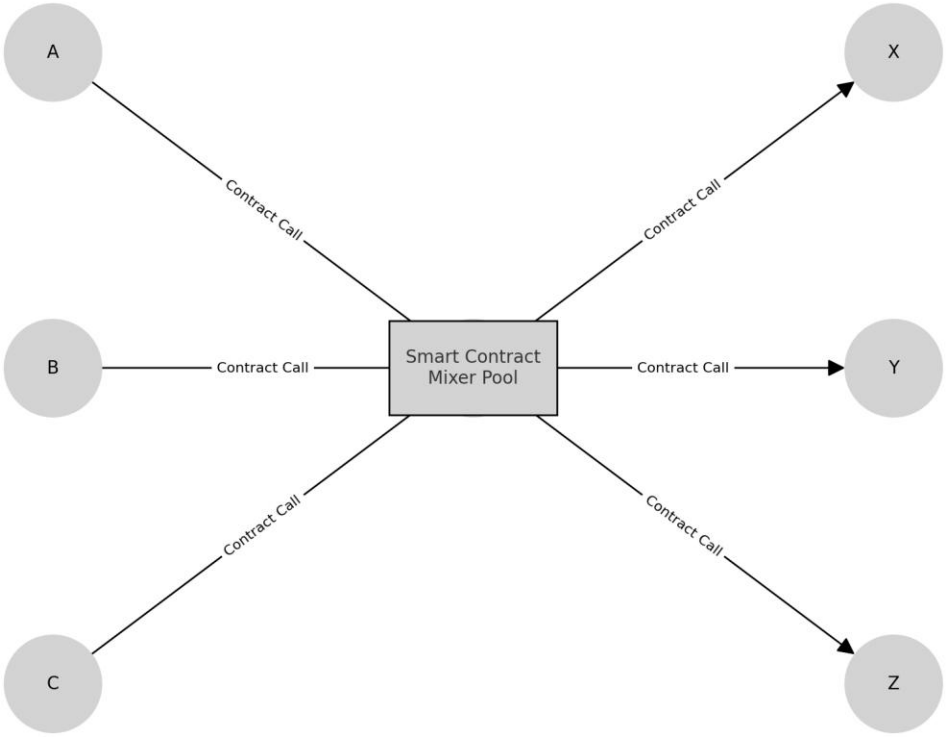

Figure 3 A high Overview of the Model to Decouple Users' Transactions

The core idea is to collect smart contract transactions from various users into a common pool. Doing so makes individual transactions indistinguishable, making it computationally infeasible to link a specific transaction to a particular user. This aggregation anonymizes the caller's identity and the parameters and state changes associated with the smart contract execution.

Users' identities are decoupled from transactions to enhance privacy using off-chain mechanisms. Users submit their transaction data to an off-chain transaction pool via secure channels, such as encrypted messaging or secure APIs, without broadcasting their blockchain addresses. The process described in Figure 4 Involves:

1. Secure Submission: Users encrypt their transaction details and send them to the off-chain pool manager without revealing their public keys or addresses.
2. Identity Obfuscation: The off-chain pool assigns a temporary identifier to each transaction, decoupling it from the user's blockchain identity.
3. Aggregation: Transactions are collected until certain criteria are met (e.g., a predefined number of transactions or time interval), increasing the anonymity set.

The direct link between the user's wallet address and the transaction is severed by handling the initial submission off-chain, significantly enhancing privacy.

Off-chain transaction pools act like proxies (see Figure 5) for user transactions before submitting them to the blockchain. Once the pooling criteria are met, the transactions are handed over to relayers and executors for on-chain submission and execution.

RESEARCH ARTICLE

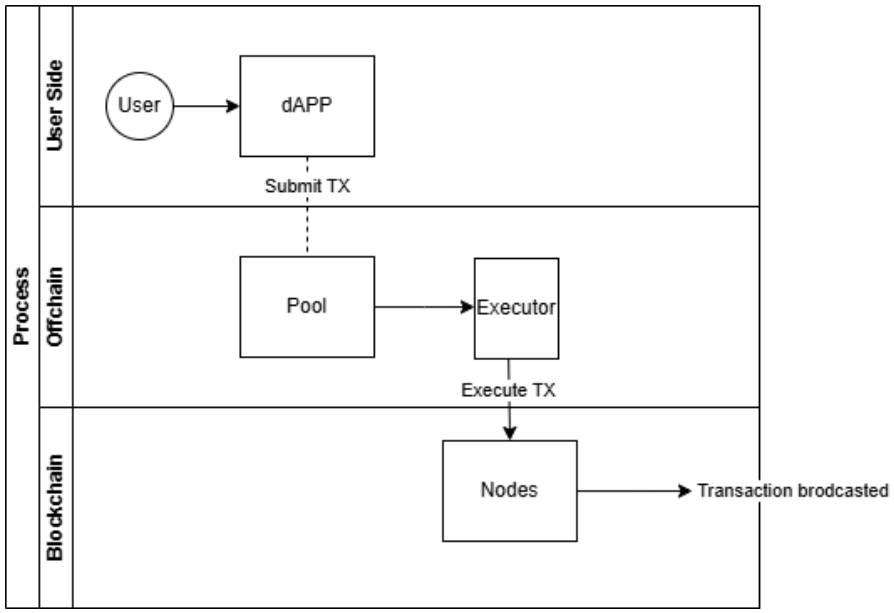

Figure 4 Relation Between the User, the Off-Chain Component, and the Blockchain Layer

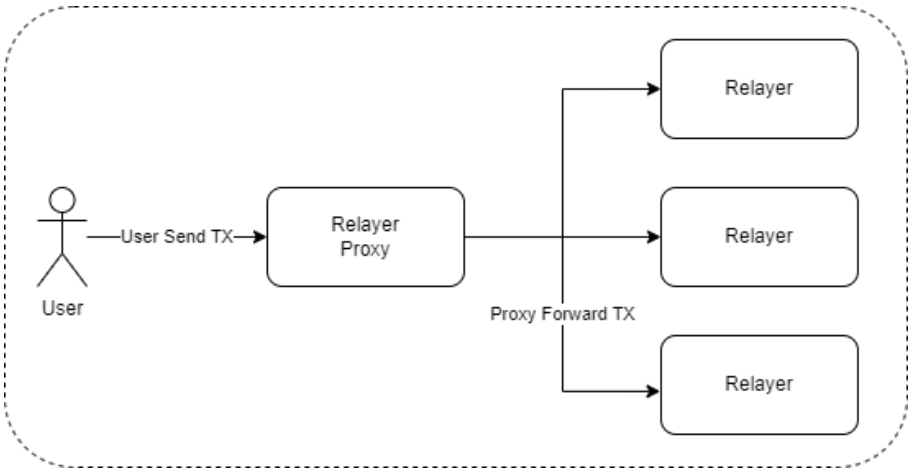

Figure 5 The Architecture of Layers in the Model

3.2. Anonymity Set and Security Analysis

Pooling transactions off-chain increases the anonymity set—the group of potential users associated with any given transaction. As the size of the pool grows, it becomes increasingly difficult for an observer to link a specific transaction to a particular user. This enhancement is due to:

- Transaction Indistinguishability:

Transaction indistinguishability is a critical mechanism in the framework that ensures all transactions within a batch appear identical to external observers, significantly enhancing privacy. By making similar transactions indistinguishable, adversaries are prevented from using transaction-specific characteristics to trace or identify individual users. In traditional blockchain systems, each transaction is visible on the public ledger, and its specific details—such as gas fees, transaction size, or the type of operation (e.g., token transfer, smart contract execution)—can reveal patterns that adversaries might exploit [18]. For example, if a particular user consistently performs transactions of a certain size or with a distinctive signature, an observer could use these characteristics to identify that user's activity. This is especially true in systems without transaction batching, where each transaction is processed individually, exposing more metadata that could be linked to a user. To counteract this, the framework uses a batching mechanism to collect and process multiple similar transactions. By grouping transactions of the same type, such as smart contract executions or token transfers, into a single batch, the transactions are

RESEARCH ARTICLE

homogeneous. From the perspective of external observers, all the transactions in a batch appear identical, effectively obfuscating any distinguishing details. This significantly reduces the risk of de-anonymization by eliminating the transaction-specific data that could be used to track a user. Each transaction within the batch is assigned a type identifier (ID), which categorizes the transaction based on the type of operation being performed, such as token transfers, contract executions, or specific protocol interactions.

- Temporal Obfuscation:

Temporal obfuscation is a critical privacy-enhancing technique that introduces variability in the timing of transaction submissions and executions. This technique helps prevent adversaries from using time-based patterns to de-anonymize users or identify the source of specific transactions. Without temporal obfuscation, an observer could monitor the precise timing of transaction execution on the blockchain and correlate it with external data or user activity to infer the identity of the transaction initiator. This vulnerability is known as a timing side-channel attack [19].

In such attacks, the adversary attempts to link a transaction's execution time to a particular user's address by analyzing patterns in the sequence and timing of transaction submissions. For example, if multiple transactions are executed quickly from a known wallet or there is a predictable pattern in the submission times, the adversary could deduce the caller's identity. To mitigate this risk, the framework employs temporal obfuscation by introducing randomness in the submission and execution times of transactions. This ensures that even if an observer monitors the blockchain in real time, the variability in execution timing makes it significantly more difficult to correlate any transaction with a specific user or their blockchain address. By adding unpredictability, the temporal patterns scrambled that might otherwise be used to trace back transactions to participants. The degree of temporal obfuscation can be dynamically adjusted based on various factors, such as the size of the anonymity set and network conditions. Importantly, the number of participating nodes in the protocol directly influences the level of obfuscation required. When fewer nodes participate in the protocol, there is a higher risk that a transaction's timing could be linked to a particular node or user. In such cases, the system must impose longer delays or wait for a larger batch of transactions to ensure that temporal patterns remain obscured. Conversely, in systems with larger nodes, the sheer volume of transactions and interactions naturally increases anonymity, reducing the need for longer waits.

To quantify the anonymity provided by the pooling mechanism, the following variables are introduced:

- N: Total number of transactions in the pool.
- U: Number of unique users submitting transactions.
- T : The time interval for which the pool collects transactions.
- $P(L)$: Probability of linking a transaction to a specific user.

If each user submits one transaction per pooling interval and all transactions are equally likely, the probability of correctly linking a transaction to a user is given by equation (1):

$$P(L) = \frac{1}{N} \quad (1)$$

As N increases, $P(L)$ approaches zero, enhancing anonymity.

However, if users submit multiple transactions or if additional information is available, more complex models, such as entropy-based measures, can be applied to assess anonymity levels.

3.3. Role of Relayers

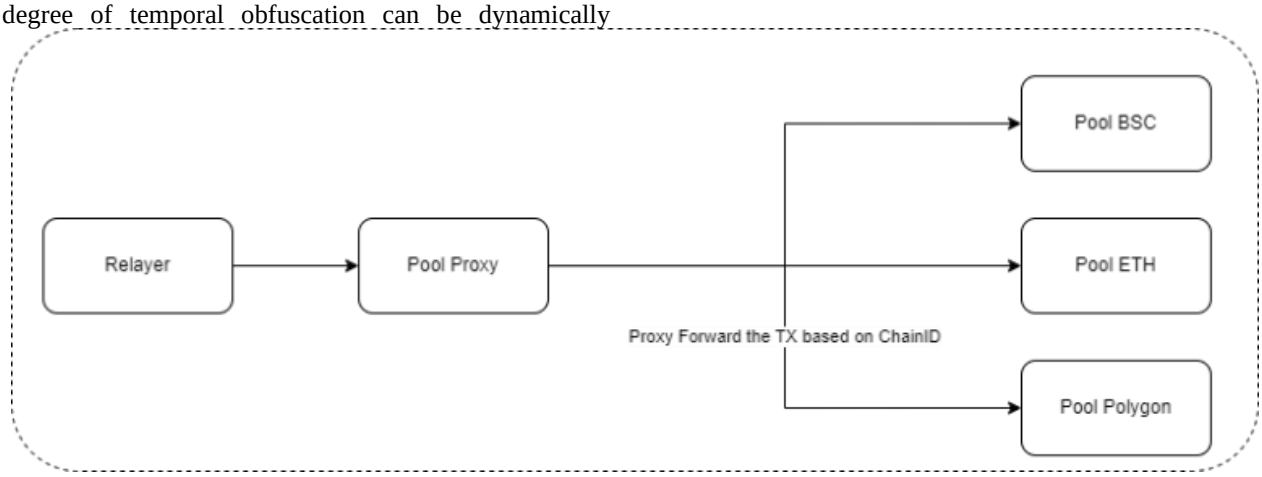

Figure 6 The Pool Proxy Forwards the Transaction to the Target Pool Chain

RESEARCH ARTICLE

Relayers are entities responsible for submitting the pooled transactions to the blockchain on behalf of users. They act as intermediaries between the off-chain transaction pool and the blockchain network (see Figure 6). The steps involved are:

1. Transaction Reception: Relayers receive encrypted transactions from the off-chain pool.
2. Aggregation: They bundle transactions into a single or multiple batches, depending on network conditions and optimization strategies.
3. Submission: Relayers submit the transactions to the blockchain, paying the necessary gas fees upfront.

Using relayers obscures the direct association between users and their transactions. Since relayers use their blockchain addresses to submit transactions, observers cannot link them back to the original users based solely on on-chain data. This mechanism ensures:

- Address Shielding: Users' wallet addresses are not exposed during the transaction submission.
- Network-Level Privacy: Relayers prevent network-level analysis from revealing user identities based on transaction propagation patterns.

3.4. Role of Executors

Executors are responsible for executing the batched transactions once they are on-chain. Their roles include:

1. Transaction Verification: Executors verify the transaction data using zero-knowledge proof verifications.
2. Smart Contract Execution: They collectively execute smart contract calls, ensuring the transactions are processed correctly.
3. State Updates: Executors update the smart contract states as per the execution results, maintaining the integrity of the blockchain ledger.

Executing multiple transactions collectively offers several performance benefits:

- Gas Optimization [20]: Batching transactions can reduce the total gas consumed per transaction by sharing fixed costs among multiple transactions.
- Throughput Improvement: Collective execution can improve the network's throughput by reducing the number of individual transactions processed.
- Latency Reduction: While there is an initial delay in collecting transactions, optimized processing can reduce the overall execution time per transaction.

The diagram below (Figure 7) shows the full system architecture, starting with the user, passing through the relayer, and finally ending with the executor, who broadcasts the transaction to the network.

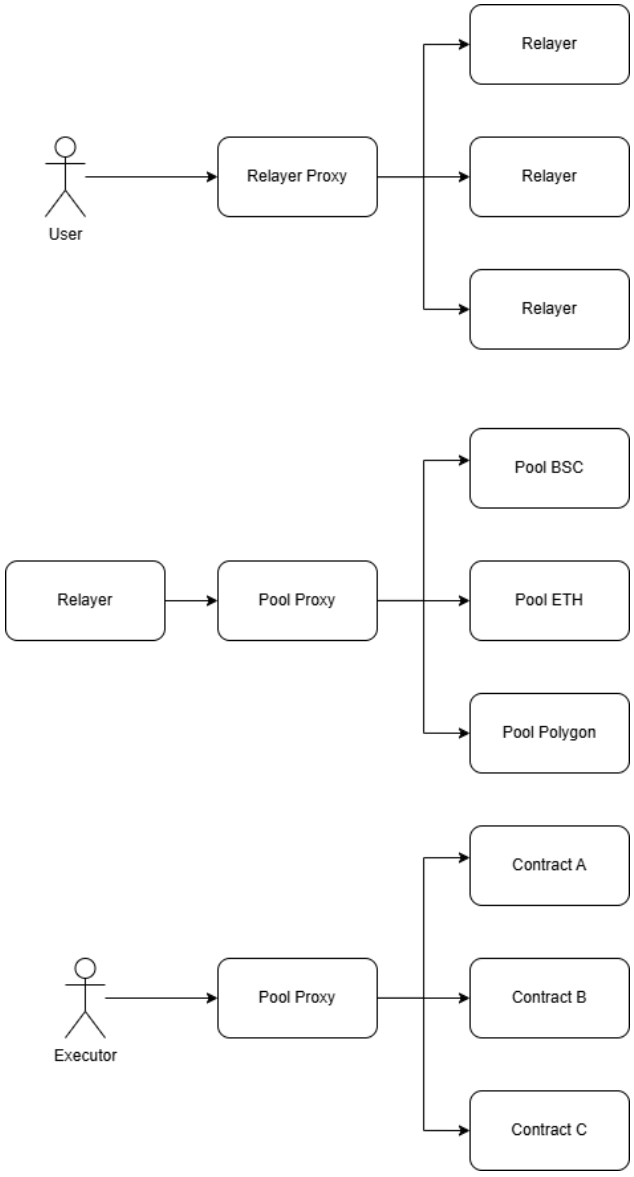

Figure 7 The Full Architecture of the Proposed Model

3.5. Incentive Mechanisms

A fair and sustainable incentive model is proposed to foster active and continuous participation from relayers and executors in the privacy-enhancing framework. The framework's success relies heavily on these participants, who play critical roles in ensuring the privacy and efficiency of smart contract executions. Since relayers are responsible for submitting transactions to the blockchain and incurring gas fees, and executors handle the computationally intensive tasks of processing and verifying transactions, they must adequately compensate for their efforts. [21]. The incentive model

RESEARCH ARTICLE

ensures that their participation remains economically viable while maintaining the network's security and performance.

3.5.1. Relayer Incentives

- Fee Compensation

One of the most fundamental aspects of relayer incentives is covering the gas fees they pay to submit transactions on the blockchain. Gas fees, which fluctuate depending on network congestion and demand [22], represent a significant cost for relayers. Therefore, the system compensates them for these fees, ensuring they do not incur out-of-pocket expenses when facilitating transactions. Additionally, the compensation model includes a profit margin to make relaying transactions financially attractive. This profit margin incentivizes relayers to prioritize and efficiently submit transactions, ensuring that the network remains functional and that privacy is maintained at scale.

- Service Fees

Beyond covering gas fees, each transaction processed by a relayer includes a small service fee payable upon successful submission. This fee ensures that relayers are rewarded for the gas they pay and the service they provide in batching and submitting transactions. By incorporating a service fee, the relayer's interests are aligned with the network's overall health [23]. Relayers will be motivated to submit transactions accurately and promptly in exchange for compensation. The service fee model also enables flexibility in designing reward structures that accommodate different types of transactions, from simple token transfers to more complex smart contract executions.

- Dynamic Pricing

To account for network activity and gas price variations, a dynamic pricing model is introduced that adjusts the compensation for relayers based on real-time conditions. In periods of high network congestion or elevated gas prices, relayers will be compensated with higher fees to reflect the increased cost of submitting transactions. Conversely, during periods of low activity, payments may be reduced to reflect lower operational costs. The dynamic pricing model also takes into consideration the urgency of transactions. For transactions that require immediate execution, relayers may charge a higher fee to prioritize their submission. This creates a market-driven mechanism where the price for relayer services adjusts fluidly based on network conditions and user demand.

3.5.2. Executor Incentives

Executors play a vital role in processing and validating transactions once they are submitted to the blockchain. Unlike relayers, who primarily handle transaction submissions, executors perform the computational work required to verify, validate, and execute smart contract transactions. Given the resource-intensive nature of this work, executors must be incentivized to contribute their computational power to the network.

- Execution Rewards

Executors are compensated through execution rewards, which are proportional to the computational effort required to process each batch of transactions. This reward structure ensures that executors are paid fairly for their work, whether it involves simple contract executions or more complex calculations. The rewards are based on several factors, including the complexity of the transactions and the gas consumption involved in their execution. More computationally demanding transactions will generate rewards, motivating executors to process standard and high-complexity batches.

- Performance Bonuses

In addition to execution rewards, executors can earn performance-based bonuses for processing transactions efficiently and within specific timeframes. This incentivizes executors to prioritize speed and accuracy, ensuring that transactions are executed on time without compromising the privacy or integrity of the network. Bonuses may be awarded for processing high volumes of transactions, completing batches before a certain deadline, or resolving particularly complex transaction batches. By offering additional incentives for exceptional performance, the system encourages executors to maintain high standards of operation and efficiency.

- Fairness and Scalability

The incentive model for executors also ensures fair compensation in a scalable manner. As the number of transactions the network processes increases, the reward structure adapts to maintain economic balance, ensuring that executors remain fairly compensated even as demand fluctuates. This scalability is particularly important as the network grows, allowing for increasing transactions and participants without diluting the value of rewards for executors. The system is designed to scale proportionally, ensuring that rewards remain consistent with the work done, regardless of the network’s size.

3.6. Sustainability and Game Theory

The incentive model is designed using game theory principles to ensure long-term sustainability. It ensures that all participants—relayers, executors, and users—act in a manner that benefits the overall network. The system is structured to reach a Nash equilibrium. [24], where the optimal strategy for each participant is to act honestly and efficiently. The system's integrity is maintained by providing clear incentives

RESEARCH ARTICLE

for relayers and executors to behave in ways that promote privacy and security.

- Nash Equilibrium

In the model, relayers and executors are incentivized to act honestly because deviation from these behaviors (e.g., delaying transaction submissions or incorrectly executing batches) would lead to a loss of profit. The reward structure is calibrated so that the most profitable strategy for participants aligns with acting in the network's best interests. By doing so, the participants naturally contribute to maintaining the integrity and performance of the system.

- Profit Maximization

Participants aim to maximize their profits while contributing to the privacy and efficiency of the network. The incentive model is structured to reward participants for maximizing network throughput and minimizing latency, balancing the needs of relayers and executors while maintaining the system's confidentiality.

- Cost-Benefit Analysis for Users

The incentive system also allows users to weigh the cost of additional transaction fees against the benefits of increased privacy. Users who prioritize privacy may be willing to pay higher fees for services provided by relayers and executors, while users requiring faster or immediate execution may opt for lower privacy levels and reduced costs. This flexibility encourages greater system adoption by catering to different user preferences, ensuring the network remains attractive to a broad range of participants.

3.7. Integration of Zero-Knowledge Proofs

Zero-knowledge proofs (ZKPs) play a pivotal role in the framework by enabling the verification of transaction validity without exposing sensitive data. The power of ZKPs lies in allowing one party (the prover) to convince another party (the verifier) that a particular statement is true without revealing any additional information about the underlying data. This concept is especially important in blockchain environments, where transparency and privacy must coexist, allowing transactions to be verified without compromising user confidentiality.

In the framework, ZKPs are employed at multiple stages of the transaction process, ensuring user authentication and transaction integrity while safeguarding sensitive details. The seamless integration of ZKPs enhances privacy without hindering the functionality or scalability of the blockchain network.

3.7.1. User Authentication with Zero-Knowledge Proofs

One of the primary applications of ZKPs in the system is to enable user authentication without revealing any identifying information about the user. [25]. Traditionally, proving ownership of assets or authorization to execute a transaction on the blockchain requires identity verification, which can expose user information. In contrast, with ZKPs, users can cryptographically prove they possess the right to execute a transaction without disclosing their public key, wallet address, or other sensitive details.

The process works as follows: a user generates proof demonstrating that they are authorized to perform a specific action, such as calling a smart contract or initiating a transfer, without revealing the underlying data. This protects the user's privacy while maintaining the transparency needed for secure transaction processing.

Thus, zero-knowledge proofs effectively decouple the proof of rights from the user's identity. This ensures that even though the transaction can be validated, the verifier (such as a relayer or a validator) gains no insight into who initiated the transaction.

3.7.2. Ensuring Transaction Integrity

Another critical function of ZKPs in the framework is to maintain transaction integrity. Validators, the parties responsible for confirming transactions on the blockchain, must ensure that all necessary criteria for executing a transaction are met, such as providing funds available or verifying that the contract conditions are satisfied. However, direct access to the transaction details could expose sensitive information, which is undesirable in a privacy-focused system.

ZKPs enable validators to confirm that a transaction adheres to the rules without viewing the underlying data. For instance, zero-knowledge proof could be used to demonstrate that a transaction's conditions have been met (e.g., sufficient funds or proper authorization) without revealing the exact amount of funds or the specific logic of the smart contract.

Using ZKPs, validators can verify that the transaction is valid and secure while ensuring that no private information is leaked.

3.7.3. Mathematical Formulations

Integrating ZKPs into the framework involves several core mathematical principles underpinning proof generation, verification, and security.

1. Statement Definition:

To initiate the zero-knowledge proof process, the user (or prover) must define a statement S That asserts the truth of a particular claim. For instance, a user could define S as "I possess sufficient funds to execute this transaction" or "I have the authority to invoke this smart contract." This statement must be formulated to be proven without directly revealing the underlying data associated with the claim.

RESEARCH ARTICLE

2. Proof Generation

Using cryptographic algorithms, the user generates a proof πThat supports the statement S. The proof demonstrates that the user knows a secret. w (such as a private key or other sensitive information) such that the statement S holds. Importantly, the proof is constructed to reveal no details about the secret itself. This is the key to maintaining privacy while ensuring the claim's validity.

3. Verification

Once the proof π is generated, it is submitted to the validator, who then applies a verification function $V(S, \pi)$ to determine whether the proof is valid. The verification process ensures that S holds without revealing the secret w. In other words, the validator can be confident that the prover is authorized to execute the transaction. Still, the prover's identity or specific transaction data remains hidden.

3.8. Compatibility and Integration with Existing Blockchains

The framework is designed to operate entirely at the application layer, ensuring it can seamlessly integrate with existing blockchain systems without requiring changes to the underlying protocol. This design choice is critical for promoting the wide adoption of privacy-enhancing technologies while maintaining the integrity and stability of blockchain infrastructures. By avoiding the need for protocol-level modifications, such as altering consensus mechanisms or network rules, the framework ensures that blockchain platforms can continue functioning as intended while benefiting from enhanced privacy features. This approach also provides developers with a low-risk, high-impact solution that simplifies integration across various platforms.

3.8.1. Application-Layer Focus

The primary advantage of focusing on application-layer functionality is that it allows the framework to be implemented on top of existing blockchain networks without disrupting their core operations. This means there is no need for complex upgrades, hard forks, or consensus changes, which are often required when making modifications at the protocol level. These changes can be risky and controversial, as they may introduce unforeseen vulnerabilities or divide the network. By confining the privacy-enhancing features to the application layer, the developers can incorporate them into decentralized applications (dApps) and smart contracts with minimal effort and no need for extensive network-wide upgrades. This simplicity is a key selling point for developers and platform operators alike, as it allows for incremental adoption without imposing additional costs or risks on the broader ecosystem. Furthermore, this approach maintains blockchain networks' inherent decentralization and security, as the underlying protocol remains untouched.

For example, integrating the framework into an Ethereum-based dApp would involve implementing privacy-enhancing features at the smart contract level without modifying Ethereum's proof-of-work or proof-of-stake consensus mechanisms. This modularity provides flexibility and reduces the technical burden for developers looking to incorporate enhanced privacy into their applications.

3.8.2. Blockchain Agnosticism

Another crucial aspect of the framework is its blockchain agnosticism, which is not confined to a single blockchain platform or protocol. While the solution leverages the widely adopted Ethereum Virtual Machine (EVM) for ease of deployment, it is designed to be compatible with a variety of blockchain ecosystems that support smart contract functionality, including Binance Smart Chain (BSC), Polkadot, Avalanche, Polygon, and others.

By maintaining compatibility with EVM, the developers can use Ethereum's extensive tooling and developer ecosystem, including popular libraries, development environments, and testing frameworks. However, the framework is flexible enough to be extended to other blockchains that may operate with different consensus algorithms or virtual machines. For example, blockchains like Polkadot, which uses a nominated proof-of-stake (NPoS) consensus model [26], or Cosmos, which uses the Tendermint consensus protocol [27], can still benefit from the privacy-enhancing features of the framework.

This cross-platform compatibility is essential in the rapidly evolving blockchain space, where new platforms and technologies are continually emerging. The framework's blockchain-agnostic design allows it to remain relevant and adaptable as the landscape changes, ensuring that developers on various platforms can implement privacy protections without being constrained by the specifics of any single blockchain.

3.9. Potential for Further Expansion

While the framework addresses the critical need for privacy in smart contract executions, several avenues for further exploration and enhancement could significantly improve its capabilities and broaden its applications.

- Dynamic Anonymity Sets

Optimizing anonymity sets based on real-time network conditions is a promising expansion area. Currently, the size and composition of the anonymity set (i.e., the pool of users among whom an individual's transaction is indistinguishable) are determined statically or periodically. However, as network traffic and transaction volumes fluctuate, there may be opportunities to dynamically adjust the anonymity set size in real time to optimize privacy. The framework could adjust the anonymity set by developing algorithms that monitor network conditions—such as transaction frequency, gas prices, or the

RESEARCH ARTICLE

number of active users. During periods of high activity, smaller batches may be sufficient to provide strong privacy guarantees, while during quieter times, larger batches may be necessary to maintain the same level of anonymity. This dynamic approach would not only enhance privacy but also improve efficiency by reducing unnecessary delays in transaction processing.

- Enhanced Cryptographic Techniques

Another important area for future development is the exploration of post-quantum cryptography to future-proof the framework against emerging threats, particularly those posed by quantum computing. While current cryptographic techniques, including zero-knowledge proofs (ZKPs), are highly secure against classical computing attacks, the advent of quantum computers could potentially undermine these security assumptions. By incorporating post-quantum cryptographic techniques [28] into the framework, it remains secure even in the face of advances in quantum computing. Techniques such as lattice-based cryptography [29] or hash-based signatures could be integrated into the framework to provide quantum-resistant proofs and signatures, ensuring that privacy protections remain robust for decades.

- Cross-Chain Compatibility

Developing as the blockchain ecosystem becomes increasingly interconnected, with multiple networks interacting through cross-chain bridges and interoperability protocols, there is a growing need for privacy solutions that can operate across different blockchains. Developing cross-chain compatibility would allow the privacy framework to be used in multi-chain environments, enabling private smart contract executions that span multiple networks. For example, a user could initiate a private transaction on Ethereum, which triggers a related action on Binance Smart Chain or Polkadot. Ensuring that the user's privacy is maintained across all participating chains would require the development of interoperability protocols that preserve privacy while allowing for secure communication between blockchains.

4. RESULTS

To evaluate the effectiveness of the proposed framework in enhancing privacy, a theoretical analysis of the anonymity set size and its impact on user anonymity was conducted. The anonymity set refers to the group of users among whom an individual's transaction is indistinguishable. A larger anonymity set implies greater difficulty for an adversary linking transactions to specific users.

4.1. Relationship Between Pool Size and Anonymity

The anonymity level A is defined as entropy, which measures the uncertainty associated with identifying a user within the anonymity set. Assuming each user is equally likely to have initiated any transaction in the pool, the entropy H can be calculated as shown in the equation (2):

$$A = H = -\sum_{i=1}^{N} p_i \log_2(p_i) \quad (2)$$

Since $p_i = \frac{1}{N}$ for all users in a uniformly random distribution, the anonymity level simplifies to equation (3):

$$A = \log_2(N) \quad (3)$$

This logarithmic relationship indicates that the anonymity level grows as the pool size increases but with diminishing returns.

Table 1 Anonymity Set Size vs. Probability of Transaction Linkage

Pool Size (N)	Anonymity Level	Probability of Linkage
10	3.32	0.1
100	6.64	0.01
1000	9.97	0.001
10000	13.29	0.0001

As shown in Table 1, increasing the pool size from 10 to 10,000 reduces the linkage probability from 10% to 0.01%, while the anonymity level increases from approximately 3.32 bits to 13.29 bits.

4.2. Impact of Multiple Transactions per User

The framework's impact on network load and efficiency is largely determined by how it handles transaction batching, which can influence transaction throughput and overall network performance. This analysis explored the effects of batching on transaction throughput and identified the trade-offs between efficiency and latency.

In traditional blockchain systems, transactions are processed individually, and the network's throughput is limited by its maximum transactions per second (TPS). Each transaction incurs a fixed computational overhead, and as the number of transactions increases, this overhead can significantly affect network efficiency. In contrast, the proposed framework processes transactions in batches, reducing the per-transaction overhead and potentially increasing throughput.

To understand this, the time to process transactions in both the traditional and the proposed batch-based approaches is modeled. Let $T_{individual}$ represent the time to process a single transaction and T_{batch} represents the time to process a batch of N transactions. Each transaction incurs a fixed overhead. O

RESEARCH ARTICLE

and a variable processing time V. Thus, the time to process a single transaction is given by the equation (4):

$$T_{individual} = O + V \quad (4)$$

In the batch processing model, the time to process a batch of N transactions is given by equation (5):

$$T_{batch} = O + N \times V \quad (5)$$

The per-transaction processing time in this model then becomes the equation (6):

$$T_{per\_transaction} = \frac{T_{batch}}{N} = \frac{O}{N} + V \quad (6)$$

As the batch size N increases, the term $\frac{O}{N}$ decreases, leading to a reduction in per-transaction processing time. This decrease in overhead per transaction is the key driver for increasing throughput in the proposed framework.

4.3. Latency Considerations

However, batching transactions introduces a trade-off between latency and efficiency. While batching reduces computational overhead and increases throughput, it can also lead to delays, as transactions must wait until enough transactions are available to form a batch. This delay could be problematic for users requiring immediate execution, especially in time-sensitive transactions.

The benefits of batching include improved efficiency, as the computational resources are shared across multiple transactions, and reduced gas fees, as the fixed costs are distributed. On the downside, users may experience increased waiting times while transactions are pooled into batches, making this approach less suitable for those requiring real-time processing.

Optimization strategies such as dynamic pooling can mitigate this trade-off. This strategy adjusts batch sizes and thresholds based on network activity, balancing latency with throughput efficiency. Furthermore, users could be given the option to choose between immediate execution (at the cost of reduced privacy) and delayed execution (with enhanced privacy) depending on their preferences and the nature of their transactions.

Despite this trade-off, the overall increase in throughput and the boost in anonymity are significant advantages that set the proposed approach apart from traditional per-transaction methods. Batching shares overhead costs among multiple users and masks individual senders within large pools, making deanonymization more difficult. A comparison of how this framework measures up against other known privacy approaches (e.g., Tornado Cash, Zcash, and Monero) is illustrated in Table 2. Key criteria include scope of anonymity, support for smart contract operations, and performance overhead.

Table 2 Comparison of the Proposed Framework with Other Privacy Solutions

Method	Tornado Cash	Zcash	Monero	Proposed Framework
Smart contract support	Limited (Token only)	None	None	Yes
Anonymity Technique	Zero-knowledge proof	Zk-SNARKS	Ring Signature	Offchain batching+ ZKPs
Overhead	Medium	Medium to High	Medium	Low to moderate
Scalability	Limited by gas	Limited by L1	Medium	High
Trusted Setup	Yes	Yes	No	No

Tornado Cash achieves anonymity by mixing tokens on-chain but does not protect complex smart contract calls.

Zcash and Monero both focus on private monetary transactions as standalone cryptocurrencies. They offer strong anonymity but lack support for Ethereum-based smart contract interactions.

The proposed Framework operates entirely at the application layer, anonymizes more complex operations (like smart

RESEARCH ARTICLE

contract executions), and leverages off-chain transaction batching to reduce overhead and enhance scalability.

As seen above, the proposed model's primary advantage lies in handling contract-level interactions while maintaining low overhead through off-chain pooling. This contrasts with on-chain mixers, which typically impose higher fees or longer confirmation times. Unlike some other zero-knowledge-based solutions, the proposed framework does not require changes to the underlying blockchain protocol or trust in a single setup ceremony.

Overall, this approach offers a high degree of flexibility: users can either prioritize reduced latency (by submitting transactions immediately) or maximize privacy and cost savings (by waiting for larger batches). Combining anonymity, scalability, and compatibility with existing blockchain platforms makes the model a strong candidate for broader adoption in privacy-critical decentralized applications.

5. CONCLUSION

This research presents a framework that significantly enhances privacy in blockchain smart contract executions. By extending the principles of transaction mixing from Tornado Cash to smart contract interactions, the framework successfully anonymizes the identities of users engaging with smart contracts. The integration of off-chain transaction pools, relayers, and executors decouples user identities from their transactions, making it exceedingly difficult for external observers to link actions to specific individuals. Using zero-knowledge proofs (ZKPs) and multi-party computation (MPC) further fortifies privacy by ensuring that transactions are verifiable and executable without revealing sensitive information. The theoretical analyses and case studies demonstrate that the probability of linking a transaction to a user decreases substantially as the anonymity set grows, confirming the framework's effectiveness in enhancing privacy.

REFERENCES

[1] Zheng, Z., Xie, S., Dai, H., Chen, X., Wang, H.: An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. In: 2017 IEEE International Congress on Big Data (BigData Congress). pp. 557–564 (2017). https://doi.org/10.1109/BigDataCongress.2017.85.

[2] Sedlmeir, J., Lautenschlager, J., Fridgen, G., Urbach, N.: The transparency challenge of blockchain in organizations. Electron Markets. 32, 1779–1794 (2022). https://doi.org/10.1007/s12525-022-00536-0.

[3] Biryukov, A., Tikhomirov, S.: Deanonymization and Linkability of Cryptocurrency Transactions Based on Network Analysis. In: 2019 IEEE European Symposium on Security and Privacy (EuroS&P). pp. 172–184 (2019). https://doi.org/10.1109/EuroSP.2019.00022.

[4] Beres, F., Seres, I.A., Benczúr, A.A., Quintyne-Collins, M.: Blockchain is Watching You: Profiling and Deanonymizing Ethereum Users. In: 2021 IEEE International Conference on Decentralized Applications and Infrastructures (DAPPS). pp. 69–78 (2021). https://doi.org/10.1109/DAPPS52256.2021.00013.

[5] Pillai, A., Saraswat, V., Vasanthakumary Ramachandran, A.: Attacks on Blockchain Based Digital Identity. In: Prieto, J., Partida, A., Leitão, P., and Pinto, A. (eds.) Blockchain and Applications. pp. 329–338. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-030-86162-9_33.

[6] Tatar, U., Gokce, Y., Nussbaum, B.: Law versus technology: Blockchain, GDPR, and tough tradeoffs. Computer Law & Security Review. 38, 105454 (2020). https://doi.org/10.1016/j.clsr.2020.105454.

[7] Pertsev, A., Semenov, R., Storm, R.: Tornado Cash Privacy Solution Version 1.4, https://crebaco.com/planner/admin/uploads/whitepapers/2982941Tornado.cash_whitepaper_v1.4.pdf, (2019).

[8] De Santis, Alfredo, and Giuseppe Persiano. "Zero-knowledge proofs of knowledge without interaction." In Proceedings., 33rd Annual Symposium on Foundations of Computer Science, pp. 427-436. IEEE Computer Society, 1992.

[9] Tang, Y., Xu, C., Zhang, C., Wu, Y., Zhu, L.: Analysis of Address Linkability in Tornado Cash on Ethereum. In: Lu, W., Zhang, Y., Wen, W., Yan, H., and Li, C. (eds.) Cyber Security. pp. 39–50. Springer Nature, Singapore (2022). https://doi.org/10.1007/978-981-16-9229-1_3.

[10] Quesnelle, J.: On the linkability of Zcash transactions, http://arxiv.org/abs/1712.01210, (2017). https://doi.org/10.48550/arXiv.1712.01210.

[11] Fujisaki, E., Suzuki, K.: Traceable Ring Signature. In: Okamoto, T. and Wang, X. (eds.) Public Key Cryptography – PKC 2007. pp. 181–200. Springer, Berlin, Heidelberg (2007). https://doi.org/10.1007/978-3-540-71677-8_13.

[12] Cao, T., Yu, J., Decouchant, J., Luo, X., Verissimo, P.: Exploring the Monero Peer-to-Peer Network. In: Bonneau, J. and Heninger, N. (eds.) Financial Cryptography and Data Security. pp. 578–594. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-51280-4_31.

[13] Du, W., Atallah, M.J.: Secure multi-party computation problems and their applications: a review and open problems. In: Proceedings of the 2001 workshop on New security paradigms. pp. 13–22. Association for Computing Machinery, New York, NY, USA (2001). https://doi.org/10.1145/508171.508174.

[14] Darby, M.L., Harmse, M., Nikolaou, M.: MPC: Current Practice and Challenges. IFAC Proceedings Volumes. 42, 86–98 (2009). https://doi.org/10.3182/20090712-4-TR-2008.00014.

[15] Fuchsbauer, G.: Subversion-Zero-Knowledge SNARKs. In: Abdalla, M. and Dahab, R. (eds.) Public-Key Cryptography – PKC 2018. pp. 315–347. Springer International Publishing, Cham (2018). https://doi.org/10.1007/978-3-319-76578-5_11.

[16] Gueron, S., Persichetti, E., Santini, P.: Designing a Practical Code-Based Signature Scheme from Zero-Knowledge Proofs with Trusted Setup. Cryptography. 6, 5 (2022). https://doi.org/10.3390/cryptography6010005.

[17] Andola, N., Raghav, Yadav, V.K., Venkatesan, S., Verma, S.: Anonymity on blockchain based e-cash protocols—A survey. Computer Science Review. 40, 100394 (2021). https://doi.org/10.1016/j.cosrev.2021.100394.

[18] Averin, A., Samartsev, A., Sachenko, N.: Review of Methods for Ensuring Anonymity and De-Anonymization in Blockchain. In: 2020 International Conference Quality Management, Transport and Information Security, Information Technologies (IT&QM&IS). pp. 82–87 (2020). https://doi.org/10.1109/ITQMIS51053.2020.9322974.

[19] Lawson, N.: Side-Channel Attacks on Cryptographic Software. IEEE Security & Privacy. 7, 65–68 (2009). https://doi.org/10.1109/MSP.2009.165.

[20] Marchesi, L., Marchesi, M., Destefanis, G., Barabino, G., Tigano, D.: Design Patterns for Gas Optimization in Ethereum. In: 2020 IEEE International Workshop on Blockchain Oriented Software Engineering

RESEARCH ARTICLE

(IWBOSE). pp. 9–15 (2020). https://doi.org/10.1109/IWBOSE50093.2020.9050163.
[21] Han, R., Yan, Z., Liang, X., Yang, L.T.: How Can Incentive Mechanisms and Blockchain Benefit with Each Other? A Survey. ACM Comput. Surv. 55, 136:1-136:38 (2022). https://doi.org/10.1145/3539604.
[22] Khan, M.M.A., Sarwar, H.M.A., Awais, M.: Gas consumption analysis of Ethereum blockchain transactions. Concurrency and Computation: Practice and Experience. 34, e6679 (2022). https://doi.org/10.1002/cpe.6679.
[23] Mssassi, S., Abou El Kalam, A.: Game Theory-Based Incentive Design for Mitigating Malicious Behavior in Blockchain Networks. Journal of Sensor and Actuator Networks. 13, 7 (2024). https://doi.org/10.3390/jsan13010007.
[24] Li, Wenbai, Mengwen Cao, Yue Wang, Changbing Tang, and Feilong Lin. "Mining pool game model and nash equilibrium analysis for pow-based blockchain networks." *Ieee Access* 8 (2020): 101049-101060.
[25] Dwivedi, A.D., Singh, R., Ghosh, U., Mukkamala, R.R., Tolba, A., Said, O.: Privacy preserving authentication system based on non-interactive zero knowledge proof suitable for Internet of Things. J Ambient Intell Human Comput. 13, 4639–4649 (2022). https://doi.org/10.1007/s12652-021-03459-4.
[26] Li, S.-N., Spychiger, F., Tessone, C.J.: Reward Distribution in Proof-of-Stake Protocols: A Trade-Off Between Inclusion and Fairness. IEEE Access. 11, 134136–134145 (2023). https://doi.org/10.1109/ACCESS.2023.3336418.
[27] Cason, D., Fynn, E., Milosevic, N., Milosevic, Z., Buchman, E., Pedone, F.: The design, architecture and performance of the Tendermint Blockchain Network. In: 2021 40th International Symposium on Reliable Distributed Systems (SRDS). pp. 23–33 (2021). https://doi.org/10.1109/SRDS53918.2021.00012.
[28] Dam, D.-T., Tran, T.-H., Hoang, V.-P., Pham, C.-K., Hoang, T.-T.: A Survey of Post-Quantum Cryptography: Start of a New Race. Cryptography. 7, 40 (2023). https://doi.org/10.3390/cryptography7030040.
[29] Nejatollahi, H., Dutt, N., Ray, S., Regazzoni, F., Banerjee, I., Cammarota, R.: Post-Quantum Lattice-Based Cryptography Implementations: A Survey. ACM Comput. Surv. 51, 129:1-129:41 (2019). https://doi.org/10.1145/3292548.

Authors

Souhail Mssassi is a Senior Security Engineer with over a decade of experience specializing in application security, cryptography, and the security of decentralized applications. He has worked with leading organizations to enhance cybersecurity strategies and secure emerging technologies. As a professor and conference speaker, Souhail has delivered lectures on security topics at universities and international events, bridging the gap between academia and industry. His current research explores formal verification and blockchain security, contributing to advancements in the reliability and resilience of modern systems.

Anas Abou El Kalam is a full professor at Marrakesh's ENSA/Cadi Ayyad University, where he heads the Cyberdefense and Embedded Telecommunication System Department and oversees the Cyberdefense, Infrastructures, and Data Protection Master's program. He is president of the Moroccan Association of Digital Trust and holds ISO 27001 Lead Auditor, CEH, and CISSP certifications. Formerly Assistant Director of the OSCARS Lab and Associate Professor at Toulouse's INP (where he earned his HDR and PhD), he has led multiple departments at ENSIB in France and trained at the Ministry of Defense in Bourges. He chairs key conferences (including the 2023 Global Summit Symposium and the 14th International Conference on Cryptology and Network Security), serves on prestigious security program committees (IEEE ACSAC, SECRYPT, IFIP SEC, ESORICS), and conducts research on blockchain-based security policies, IoT security, big data security, and critical infrastructure protection. Co-author of more than 150 research papers, he has contributed to Airbus and European projects (PRIME, CRUTIAL, NoE Newcom++, Celtic Fell@home).

How to cite this article:

Souhail Mssassi, Anas Abou El Kalam, "An Application-Layer Framework for Privacy Blockchain Transactions and Smart Contract", International Journal of Computer Networks and Applications (IJCNA), 12(1), PP: 62-76, 2025, DOI: 10.22247/ijcna/2025/05.