Enabling permissioned-to-permissionless blockchain interoperability with enhanced privacy and incentive mechanisms

by

Saeed Moradi

A thesis submitted to
Faculty of Graduate and Postdoctoral Studies
The University of Manitoba
in partial fulfillment of the requirements
of the degree of

Master of Science

Department of Computer Science
The University of Manitoba
Winnipeg, Manitoba, Canada
August 2025

© Copyright 2025 by Saeed Moradi

Thesis advisor

Sara Rouhani

Author

Saeed Moradi

Enabling permissioned-to-permissionless blockchain interoperability with enhanced privacy and incentive mechanisms

Abstract

Interoperability between blockchain networks is a critical capability for enabling autonomous blockchain systems to communicate and exchange information. Bridging a permissioned network with a permissionless network introduces unique challenges, particularly in safeguarding the privacy and anonymity of stakeholders within the permissioned network, where stricter privacy and governance requirements apply. This thesis presents a relay-based architecture that facilitates secure interoperability between these two categories of blockchains while mitigating privacy risks inherent in such cross-chain interactions. The relay layer comprises autonomous blockchain nodes that operate independently of both networks. To incentivize active and reliable participation, we propose a hybrid incentive mechanism incorporating a trust score to quantify and reward the reliability and trustworthiness of each node. The proposed mechanism is evaluated under a defined threat model, with security risks systematically identified and addressed. Finally, the architecture is assessed in terms of both performance and its effectiveness in preserving privacy and anonymity.

Contribution of Authors

Saeed Moradi is the primary author of this thesis. Sara Rouhani contributed to developing the research idea and provided minor revisions to the thesis text.

Contents

Abstract ii
Contribution of Authors iii
Table of Contents vi
List of Figures vii

1 Introduction **1**
1.1 Research Questions 3
1.2 Methodology and Approach 4
1.2.1 System Design 4
1.2.2 Threat Modeling and Security Analysis 5
1.2.3 Evaluation Strategy 5
1.2.4 Validation 6
1.3 Contributions 6
1.3.1 Interoperability Framework 7
1.3.2 Incentive and Trust Mechanism 7

2 Background **9**
2.1 Blockchain 9
2.2 Blockchain Governance 11
2.3 Smart Contracts 12
2.4 Blockchain Interoperability 14
2.5 Interoperable Blockchain Architecture 14
2.6 Anonymization 16
2.7 Source Anonymization Protocols 16
2.7.1 Dandelion++ 17
2.7.2 Clover 18
2.8 Incentive Mechanisms 19

3 Related Works **22**
3.1 Blockchain Interoperability 22
3.1.1 Interoperability between Permissionless/public Blockchains . . 23

3.1.2 Interoperability Between Permissioned Blockchains 24
3.1.3 Solutions Bridging Permissioned and Permissionless Blockchains 24
3.1.4 Privacy-Preserving Techniques in Cross-Chain Systems 25
3.1.5 Incentive Mechanisms in Blockchain Networks 26
3.1.6 Discussion and Research Gap 28

4 System Design **30**
4.1 Cryptographic Primitives 30
4.1.1 Blind Signatures 30
4.1.2 Ring Signatures 32
4.1.3 Soft-Consensus Quorum Gate 32
4.2 Incentive Mechanism 33
4.2.1 Per-epoch Observable Metrics 33
4.2.2 Per-epoch performance score 35
4.2.3 Exponential moving average trust update 36
4.2.4 Stake and Effective Weight Calculation 37
4.2.5 Weighted Quorum and Reward Distribution 38
4.2.6 Collusion Detection via Agreement Statistics 38
4.3 System Components: Overview 39
4.4 Cross-Chain Transaction Life-cycle 41

5 Security Analysis **45**
5.1 Threat Model 45
5.1.1 Spoofing 46
5.1.2 Tampering 46
5.1.3 Repudiation 46
5.1.4 Information Disclosure 47
5.1.5 Denial of Service (DoS) 47
5.1.6 Elevation of Privilege 48
5.2 Sybil, 51%, and Collusion Attacks 48
5.2.1 Sybil resistance 48
5.2.2 Weighted-majority (51%) mitigation 49
5.2.3 Collusion detection and response 49
5.2.4 Informal correctness claim 50
5.3 Source Anonymization Security Analysis 51
5.3.1 Adversarial Model 51
5.3.2 Path Compromise Probability 52
5.3.3 Effect of Local Visibility 52
5.3.4 Watcher and Auditor Oversight 53
5.3.5 Takeaways 53

6 Evaluation **55**
6.1 Prototype and Evaluation Environment 56
6.1.1 Programming Environment 56
6.1.2 Simulation Setup 56
6.1.3 Evaluation Metrics 57
6.2 Evaluation Results 57
6.2.1 Figure 6.1: Reward inequality over time (Gini coefficient) . . . 58
6.2.2 Figure 6.2: Latency CDF (anonymity vs. performance trade-off) 58
6.2.3 Figure 6.3: Parameter sweep — quorum threshold τ vs. EWMA memory α 60
6.2.4 Figure 6.4: Trust score time series 62
6.2.5 Figure 6.5: Reward histogram (final epoch) 62
6.2.6 Figure 6.6: Attacker success probability vs. adversary stake fraction 64
6.2.7 Figure 6.7: Deanonymization probability vs. ring size 65

7 Conclusion **68**
7.1 Summary of contributions 69
7.2 Key findings from the evaluation 70
7.2.1 Trust dynamics and reward fairness: 70
7.2.2 Resistance to stake- and Sybil-based attacks: 70
7.2.3 Privacy / deanonymization trade-offs: 71
7.2.4 Performance and scalability: 71
7.2.5 Equity and concentration metrics: 71
7.3 Limitations 72
7.4 Practical implications and recommendations 73
7.5 Final remarks 73

8 Future Works **75**
8.1 Scalability and Performance Optimization 75
8.2 Security Hardening and Adversarial Models 77
8.3 Privacy and Anonymization Enhancements 78
8.4 Governance and Trust Frameworks 78

Bibliography **89**

List of Figures

2.1 Dandelion++ transaction flow. Each node randomly chooses up to 2 of its outbound edges, forming an anonymity graph that determines the data recipients during the fluff phase. The graph is updated after each 10 minutes-long epoch. 17
2.2 Clover transaction flow. The transaction data is proxied probabilistically between peer nodes. Each node randomly decides whether to keep the transaction data in the proxy phase (re-proxying the transaction) or broadcast it to the entire network (diffusion phase) 18

4.1 Architecture 31
4.2 System Transaction Lifecycle 41

6.1 Reward inequality over time, measured using the Gini coefficient per epoch. The Gini rises during the initial reputation-formation phase and stabilizes thereafter, indicating a fair but incentive-aligned distribution in the steady state. 59
6.2 CDF of end-to-end message finalization latency for different anonymization protocols and cryptographic configurations. The anonymization mechanisms (Clover, Dandelion++) and cryptographic processing (ring and blind signatures) introduce modest latency shifts relative to the baseline. 60
6.3 Parameter sweep results for quorum threshold τ and EWMA memory α. Left: estimated adversary success probability under the tested model. Right: median finalization latency. The results suggest a tunable safety–liveness tradeoff and indicate a viable operating region with low adversary success and acceptable latency. 61
6.4 Time series of trust scores for all nodes over the course of the simulation. Honest nodes exhibit stable or gradually increasing trust, while malicious nodes experience a decline due to penalties, reflecting the system's ability to isolate and demote misbehaving participants. . . . 63

6.5 Histogram of cumulative rewards per node at the end of the simulation. The distribution shows that rewards are correlated with sustained positive trust scores, indicating that the reward mechanism effectively incentivizes honest participation. 64
6.6 Attacker success probability as a function of total adversarial stake. The system demonstrates strong resistance to Sybil and majority stake attacks until adversaries control a critical stake threshold, beyond which compromise probability increases sharply. 66
6.7 Deanonymization probability versus ring size in the ring signature scheme. Larger ring sizes significantly reduce the probability of correctly identifying the transaction origin, demonstrating the effectiveness of the cryptographic anonymity layer. 67

Chapter 1

Introduction

Blockchain technology has emerged as a foundational innovation for building secure, tamper-resistant, and decentralized systems. At its core, a blockchain is a distributed ledger organized as a chain of cryptographically linked blocks, where each block contains a set of validated transactions along with a reference to the preceding block. Typical implementations use cryptographic hashes to secure the linkage between a block and its predecessor Xu et al. [2019b]. Over the past decade, they have been applied across domains such as finance, healthcare, supply chain management, and the internet of things, where trust minimization, data integrity, auditability, and censorship resistance are critical.

Broadly, blockchain networks are categorized as *permissioned* or *permissionless*. Permissioned blockchains, such as Hyperledger Fabric Androulaki et al. [2018], restrict participation to a predefined set of entities and provide fine-grained access control alongside enhanced confidentiality guarantees. In contrast, permissionless blockchains permit unrestricted participation and rely on incentive-driven consensus

mechanisms—such as Proof-of-Work and Proof-of-Stake—to maintain security. However, these networks make transaction data publicly accessible and suffer from higher confirmation latencies.

The rapid proliferation of blockchain applications and the rise of both permissioned and permissionless platforms have given rise to numerous blockchain networks, each tailored to specific user groups. However, these platforms typically operate in isolation and lack inherent interoperability, preventing the direct exchange of data or assets across networks. This limitation motivates the need for *blockchain interoperability*—the capability to bridge distinct blockchain systems while respecting their native consensus mechanisms, protocols, and security and privacy policies. Existing interoperability approaches can be broadly categorized into blockchain-of-blockchains architectures, sidechains, atomic swaps, and notary schemes Barbàra and Schifanella [2023]; Frauenthaler et al. [2020]; Sun et al. [2022]. While such solutions enable cross-chain communication among permissionless blockchains, they are generally unsuitable for scenarios involving permissioned networks. Establishing a connection between a permissioned and a permissionless blockchain introduces significant challenges related to security, privacy, and anonymity. Addressing these challenges is crucial, as interoperability mechanisms are inherently vulnerable to deanonymization and Sybil attacks Fanti and Viswanath [2017]; Douceur [2002].

Apart from privacy concerns, any interoperability solution inevitably relies on certain trust assumptions. Trust is necessary to ensure honest behavior from network participants and to mitigate malicious actions, particularly when off-chain components are involved in intermediating between two heterogeneous blockchains. In de-

centralized environments, blockchain nodes typically expect to be compensated for their performance in the system. In the context of blockchain technology, this compensation is realized through an *incentive mechanism*, which defines how peers are encouraged to act honestly. Incentive mechanisms reward participants in different ways, most commonly through **monetary** or **reputation**-based schemes Ihle et al. [2023]. Monetary incentives provide direct payments in the form of assets or tokens, whereas reputation-based incentives establish trust by monitoring participant behavior and rewarding consistent compliance with system rules Ihle et al. [2023]. Such mechanisms are generally enforced by the system's consensus protocol. However, due to the strict access policies of permissioned blockchains, designing an interoperability solution that aligns with their consensus mechanisms is complex and often results in non-generic solutions.

In this thesis, we address both challenges. We propose a framework that connects a permissioned blockchain to a permissionless blockchain while explicitly accounting for privacy and anonymity requirements. Furthermore, we introduce a hybrid monetary–reputation incentive mechanism, grounded in a trust score, to ensure that participants in our interoperability framework act in accordance with the goals of the network.

1.1 Research Questions

In this thesis, we investigate answering the following research questions:

RQ1 How can a permissioned blockchain securely interoperate with a permissionless blockchain while preserving the privacy requirements of the permissioned

environment?

RQ2 What security and privacy concerns are targeted towards an interoperability system and how we can mitigate these threats?

RQ3 How can we incentivize our interoperability component to behave honestly and prevent malicious behavior?

1.2 Methodology and Approach

To address our research questions, we adopt a **design–analyze–evaluate** methodology that integrates system implementation with structured threat modeling, experimental evaluation, and robustness validation.

1.2.1 System Design

We develop a **bridge component** that serves as the interoperability layer between permissioned and permissionless blockchains.

To address **RQ1**, our design incorporates **bridge contracts** deployed on both networks to govern cross-chain message exchange. These contracts rely on cryptographic primitives such as ring signatures, blind signatures, and source-anonymization protocols (e.g., Dandelion++) to preserve sender anonymity, unlinkability, and confidentiality. The bridge nodes carry out the relaying process, while *watcher nodes* independently monitor both chains to provide attestations, increasing accountability and resilience against dishonest relays. Additionally, an *off-chain auditor* analyzes

long-term behavioral patterns and submits verifiable summaries on-chain, strengthening transparency without introducing centralization.

To address **RQ3**, we design a **trust contract** that continuously evaluates the behavior of participating nodes. It computes a dynamic trust score combining reputation (based on watcher attestations and auditor reports) with stake. This score influences relay selection, reward allocation, and penalties, thereby incentivizing honest participation, deterring malicious activity, and ensuring Sybil resistance.

1.2.2 Threat Modeling and Security Analysis

To answer **RQ2**, we construct a comprehensive threat model targeting the bridge. Following the STRIDE framework, we classify risks such as spoofing, tampering, repudiation, and denial-of-service, while extending the analysis to blockchain-specific adversarial behaviors including Sybil attacks, collusion, and weighted-majority manipulation. For each threat category, we map the mitigation strategies in our design (e.g., signature verification, quorum thresholds, randomized relay selection, and trust-based weighting). Watcher attestations and auditor reports act as additional safeguards to detect collusion and long-term performance manipulation. This structured approach ensures that the system explicitly addresses the identified risks.

1.2.3 Evaluation Strategy

We evaluate the proposed system through controlled simulations and experiments. Our evaluation spans three dimensions:

- **Performance:** Measuring end-to-end message latency, throughput, and gas

cost overhead under varying network sizes and anonymization configurations.

- **Security and Privacy:** Analyzing adversarial success probabilities under Sybil and collusion scenarios, as well as deanonymization resistance across different ring sizes and anonymization protocols.

- **Incentives and Fairness:** Studying reward distribution, trust-score evolution, and fairness metrics (e.g., Gini coefficient) to verify that the mechanism promotes equitable and honest participation.

Implementation involves smart contracts deployed in a controlled test environment, simulated bridge and watcher nodes, and cryptographic routines integrated into the relay process, ensuring that evaluation reflects both feasibility and practical constraints.

1.2.4 Validation

We validate our results through parameter sweeps, ablation studies, and sensitivity analyses (e.g., varying trust memory factor α, quorum threshold τ, and ring size r). These experiments allow us to characterize trade-offs between security, privacy, and performance, while testing the robustness of the system under adversarial and environmental variations.

1.3 Contributions

To address our research questions, we make the following key contributions:

1.3.1 Interoperability Framework

We introduce a bridge-based interoperability framework that enables secure and privacy-preserving communication between a permissioned blockchain and a permissionless blockchain, without imposing additional trust assumptions on either network. Existing interoperability approaches predominantly focus on permissionless–permissionless interactions, leaving permissioned networks largely isolated due to their restricted access policies and privacy requirements. Our design addresses this gap through a dedicated **bridging component** composed of cooperating nodes that exchange data across networks. The architecture leverages cryptographic primitives, including ring signatures for origin anonymity and blind signatures for unlinkable validation, thereby ensuring that message authenticity, integrity, and confidentiality are preserved during transfer. This approach enables permissioned network participants to share information externally while upholding their internal governance rules and privacy constraints. The proposed framework preserves sender anonymity and resists network-level traffic analysis while introducing only modest latency overhead. These guarantees were validated through simulated deanonymization experiments and end-to-end latency measurements under varying anonymization configurations. Parameter sweeps further demonstrate stable security–latency trade-offs across realistic operational settings, highlighting the adaptability of the design.

1.3.2 Incentive and Trust Mechanism

We further contribute a dynamic, reputation-aware incentive mechanism that promotes honest participation among bridging nodes while deterring Sybil attacks, col-

lusion, and majority-stake manipulation. The proposed trust model computes node weights by combining normalized stake with reputation scores derived from observed behavior, accuracy of relayed messages, and indicators of collusion. By periodically recalculating trust scores and adjusting reward distribution accordingly, the mechanism ensures that both long-term reliability and recent performance are reflected in node incentives. Misbehavior incurs proportional penalties, while consistent honest performance is rewarded, creating a self-reinforcing system that aligns individual incentives with network-wide security and reliability objectives.

Chapter 2

Background

In this chapter, we present the foundational concepts necessary for understanding the architecture of the proposed system. We begin by outlining blockchain-specific principles pertinent to this thesis, followed by a discussion of the message-signing and source-anonymization protocols incorporated into the design. Finally, we examine blockchain incentive mechanisms and their role in the proposed system architecture.

2.1 Blockchain

Given the substantial body of research on blockchain, numerous studies have sought to define this technology Kamilaris et al. [2019]; Xu et al. [2019a]; Liu et al. [2019]; Tolmach et al. [2021]; Chen et al. [2020], each emphasizing distinct aspects of its operation and design. Synthesizing these definitions, blockchain can be described as a decentralized and distributed digital ledger that records transactions in a transparent, secure, and tamper-resistant manner. Initially introduced by Nakamoto [n.d.],

its earliest application was the exchange of a digital asset known as *cryptocurrency*. These assets are transferred between participants in the form of transactions.

Due to the decentralized nature of a blockchain, a blockchain operates without a central authority to verify the authenticity of transactions. Consequently, to mitigate risks such as fraud and double-spending Pinzón and Rocha [2016], transaction validation is performed by *blockchain nodes*—distributed servers that communicate with one another to process and verify transactions. These nodes must collectively agree on the validity of each transaction, a process governed by a *consensus mechanism*.

Numerous consensus mechanisms have been proposed, including Proof-of-Work (PoW), Proof-of-Stake (PoS), Proof-of-Authority (PoA), Byzantine Fault Tolerance (BFT), and Raft Ongaro and Ousterhout [2014]; Nakamoto [n.d.]; Bano et al. [2019]. Each mechanism is tailored to the characteristics and requirements of a specific operational environment. For example, Raft is employed in Hyperledger Fabric Androulaki et al. [2018], an enterprise-oriented (permissioned) blockchain that enforces controlled access to its network. In such a trusted setting, mechanisms like PoW are unsuitable, as the participating nodes are already authenticated and do not require the computationally expensive security guarantees provided by PoW.

As a result, designing a generic consensus mechanism that operates effectively across both permissioned and permissionless blockchain environments is inherently challenging due to their differing trust models, performance requirements, and fault assumptions. To address this, we implement dedicated smart contracts that interface with our bridge component to facilitate data exchange. These contracts enforce the security and privacy policies defined by the system, while also monitoring transaction

flows, system behavior, and the operation of incentivization mechanisms.

2.2 Blockchain Governance

In contrast to blockchain technology, the study of blockchain governance remains relatively underexplored. A notable contribution is provided by Kiayias et al. Kiayias and Lazos [2022], who systematically classify the fundamental properties of blockchain governance into four principal dimensions: **Voting System**, **Security**, **Incentives**, and **Timeliness**. The authors contend that an effective governance framework must comprehensively address all these dimensions to ensure robustness and long-term viability.

According to this study, there are seven key metrics that a blockchain governance protocol should satisfy:

- **Confidentiality:** Ensures the privacy of decision-makers' inputs. It encompasses *Privacy*, which requires that inputs remain confidential, and *Coercion Resistance*, which guarantees that inputs are free from any external pressures or influences.

- **Verifiability:** Enables decision-makers to verify that their inputs have been accurately recorded and that governance outcomes have been computed correctly.

- **Accountability:** Holds decision-makers responsible for the inputs they provide to the governance system.

- **Sustainability:** Assesses whether the governance protocol provides sufficient incentives to encourage constructive system evolution and meaningful participation.
- **Pareto Efficiency:** Ensures that no alternative governance outcome can improve one participant's welfare without diminishing another's, given all decision-makers' preferences.
- **Liveness:** Reflects the system's ability to reach decisions reliably and within acceptable timeframes.

Blockchain governance refers to *set of protocols, policies, and regulations governing all aspects of blockchain management, including decision-making processes, authority structures, role definitions, network and storage administration, privacy and security measures, transparency and verifiability of actions, and accountability mechanisms such as incentivization and penalization* Liu et al. [2023]; Pelt et al. [2021].

2.3 Smart Contracts

Smart contracts are self-executing programs that automatically enforce the terms of an agreement once predefined conditions are satisfied. Operating on blockchain platforms, they ensure transparency, immutability, and trustless execution. The concept was first introduced by Szabo Szabo [1994], who described them as computerized transaction protocols that execute the terms of a contract.

A defining property of smart contracts is immutability: once deployed on a blockchain, the contract's code cannot be altered or removed. This design choice

strengthens trust and security in decentralized environments but also imposes strict requirements for careful design, thorough testing, and proactive security auditing.

The advent of blockchain technologies enabled the practical realization of smart contracts, with Ethereum's 2015 launch marking a significant milestone. Ethereum introduced Solidity, a Turing-complete programming language that supports the creation of complex decentralized applications (dApps) Alharby and Van Moorsel [2017].

Smart contracts have been applied across diverse domains such as finance, healthcare, and supply chain management, offering automation that reduces operational costs, minimizes reliance on intermediaries, and mitigates risks of fraud and human errors Mohanta et al. [2018]. However, their deployment presents several challenges.

Security vulnerabilities remain a major concern, exemplified by the 2016 DAO attack Yuan [2023], where a flaw in the contract code led to significant financial losses. Furthermore, scalability limitations and the lack of standardized legal frameworks hinder widespread adoption. The enforceability of smart contracts within traditional legal systems continues to be debated Raskin [2016].

Recent research efforts have focused on incorporating formal verification techniques, enhancing smart contract programming languages, and developing governance models that align with both technical constraints and regulatory requirements Tolmach et al. [2021].

In our proposed architecture, smart contracts are integral to managing inter-blockchain data exchange, enforcing privacy policies, applying cryptographic operations, and overseeing our incentive mechanism.

2.4 Blockchain Interoperability

Parallel to blockchain, blockchain interoperability has gained tremendous attention thanks to the growing demand for blockchain ecosystems to interact with each other. There is no universally-agreed definition for blockchain interoperability. According to Ren et al. [2023], blockchain interoperability refers to "the ability of blockchains to flexibly transfer assets, share data, and invoke smart contracts across a mix of public, private, and consortium blockchains without any changes to the underlying blockchain systems". In Monika and Bhatia [2020], however, it is described when "one platform interacts with another to use its services. The interaction can be related to the referencing of data, verification of data, and use of the computational capacity of the blockchain network".

2.5 Interoperable Blockchain Architecture

According to Hardjono et al. [2020], an interoperable blockchain architecture is "a composition of distinguishable blockchain systems, each representing a unique distributed data ledger, where atomic transaction execution may span multiple heterogeneous blockchain systems, and where data recorded in one blockchain are reachable, verifiable, and referenceable by another, possibly foreign transaction in a semantically compatible manner". In the context of our study, permissioned blockchains are not designed to be interoperable with permissionless blockchains. Therefore, we aim to design a framework that enables permissioned blockchains to communicate with a permissionless blockchain.

Key Aspects of Interoperable Blockchain Architecture

1. **Cross-Chain Communication:** Facilitates the transmission of data, assets, or commands between distinct blockchain networks. This includes mechanisms for cross-chain smart contract invocation, token transfers, and data queries [Deng et al., 2025].

2. **Protocol and Consensus Agnosticism:** Supports blockchains with varying consensus algorithms and governance models (permissioned or permissionless), ensuring that interoperability does not depend on a specific blockchain type [Dinh et al., 2019].

3. **Security and Trust Models:** Maintains the security assumptions of each participating blockchain through mechanisms such as cryptographic proofs, trusted relays, or decentralized notary schemes [Hardjono et al., 2020].

4. **Interoperability Middleware:** Acts as a bridge or translator, abstracting differences between blockchains. Examples include relays, oracles, interchain protocols, or gateways that handle messaging and verification [Belchior et al., 2022].

5. **Scalability and Efficiency:** Ensures efficient handling of inter-chain operations to avoid bottlenecks or excessive transaction fees during data or asset exchanges [Deng et al., 2025].

6. **Privacy and Compliance:** Addresses privacy concerns and compliance with regulations like the General Data Protection Regulation (GDPR), especially when interoperability exposes more data across networks [Belchior et al., 2022].

2.6 Anonymization

Anonymization refers to the irreversible process of transforming personal data of individuals so that they cannot be identified European Data Protection Supervisor [2021]. According to the General Data Protection Regulation (GDPR) GDPR2016, Anonymized data refers to information that either is not related to any identifiable individual, or personal data that has been processed in such a way that the individual can no longer be identified. Within blockchain networks, anonymization remains a critical concern for user privacy. Although Bitcoin is frequently perceived as providing anonymity, empirical research contradicts this assumption Meiklejohn et al. [2013]. Consequently, privacy-focused blockchains such as Monero and Zcash have been developed, incorporating advanced deanonymization-resistance techniques to reduce the traceability of user activities Moser et al. [2018]; Kappos et al. [2018]. This thesis presents an architecture that specifically addresses user privacy protection through anonymization, given the limited literature on anonymization in interoperable blockchain systems.

2.7 Source Anonymization Protocols

Source anonymization protocols are essential for mitigating network-level deanonymization attacks, where adversaries attempt to link the origin of a transaction to an IP address. Two representative approaches are *Dandelion++* and *Clover*, both of which aim to obfuscate the source of a transaction while ensuring eventual delivery to the blockchain network. Although they share the fundamental two-phase idea of hiding

the transaction origin before diffusion, their architectures and forwarding strategies differ.

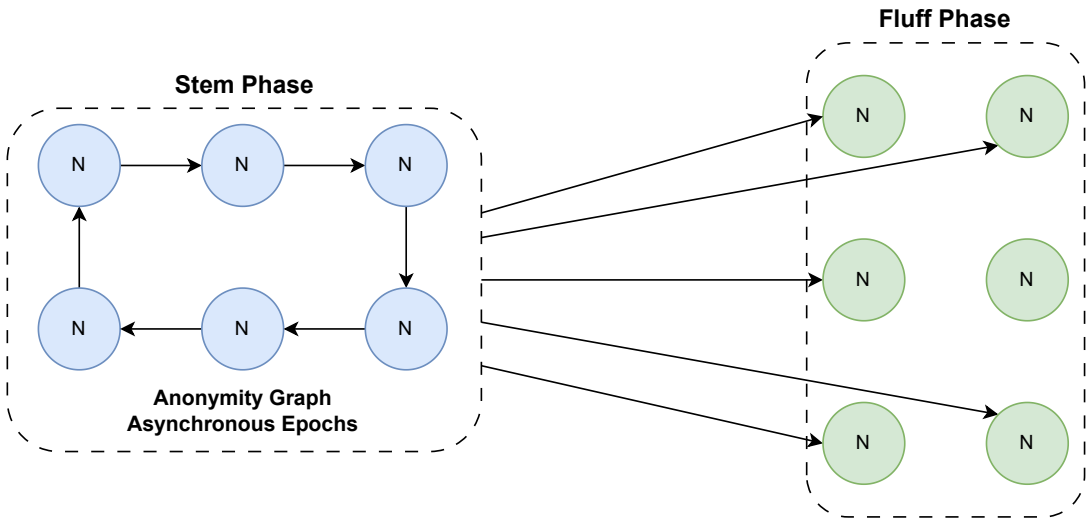

Figure 2.1: Dandelion++ transaction flow. Each node randomly chooses up to 2 of its outbound edges, forming an anonymity graph that determines the data recipients during the fluff phase. The graph is updated after each 10 minutes-long epoch.

2.7.1 Dandelion++

Dandelion++ Fanti et al. [2018] extends the original Dandelion Bojja Venkatakrishnan et al. [2017] protocol by introducing robustness against active attacks. It operates in two phases:

1. **Stem phase:** The transaction is relayed along a pseudo-random line, where each node forwards the message to exactly one neighbor, chosen deterministically but randomized across epochs. This forwarding continues for a geometri-

cally distributed number of hops.

2. **Fluff phase:** Once the stem phase ends, the last node diffuses the transaction to the entire network using standard gossip flooding. This ensures delivery and liveness.

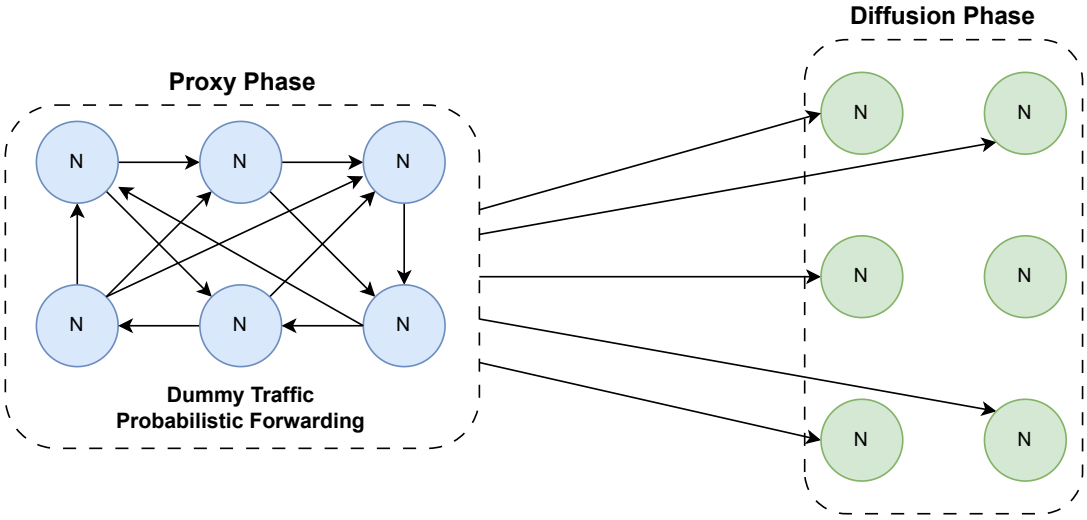

Figure 2.2: Clover transaction flow. The transaction data is proxied probabilistically between peer nodes. Each node randomly decides whether to keep the transaction data in the proxy phase (re-proxying the transaction) or broadcast it to the entire network (diffusion phase)

2.7.2 Clover

Clover Franzoni and Daza [2021] follows the same two-phase intuition but introduces **redundant paths** for robustness and stronger anonymity guarantees. Instead of forwarding the transaction along a single line during the stem phase, Clover em-

ploys multiple partially overlapping relay paths ("clovers"), which branch and later merge before entering the diffusion phase. This redundancy reduces the probability of successful deanonymization, since an adversary must control multiple paths simultaneously to link the source with high confidence.

1. **Proxy Phase:** A transaction is relayed along multiple redundant paths of controlled length, increasing resilience to blackholes and targeted intersection attacks.

2. **Diffusion Phase:** Similar to Dandelion++, the transaction eventually enters a diffusion stage where it is gossiped to the network at large.

Both Clover and Dandelion++ aim to provide *network-level anonymity* by obscuring the origin of a transaction or message within a peer-to-peer network. Clover accomplishes this through randomized routing paths, making it statistically difficult for an adversary to identify the initial broadcaster, even when the network is partially compromised. Dandelion++ uses a two-phase forwarding process to conceal the transaction source before it is broadcast to the wider network. While both protocols enhance network-level anonymity, they do not mitigate deanonymization at the cryptographic layer, which our architecture addresses through the use of ring signatures.

2.8 Incentive Mechanisms

Incentive mechanisms motivate cooperation and sustained participation by overriding participants' intrinsic self-interests through extrinsic pressures, such as severe

disincentives or rewards Ihle et al. [2023].

Incentive mechanisms can be categorized by type: **monetary**, **reputation-based**, or **service-based**. A monetary incentive encourages participants to join the network or align with its objectives by allocating financial rewards—such as digital currencies—for actions that contribute to network goals. A reputation-based incentive, first introduced in Aluko and Kolonin [2021], fosters trust by leveraging direct and indirect (transitive) observations of participant behavior to predict future actions. These observations are quantified into a reputation score, a non-transferable metric that cannot be exchanged for goods or services but serves as an indicator of trustworthiness. A service-based incentive rewards cooperation with access to specific services, resources, or privileges—such as computational power, exclusive information, or digital media. This model operates on an immediate, transactional basis: participants either gain access to the service as a reward or are excluded from it if they attempt to game the system.

In this thesis, we adopt a *monetary–reputation hybrid* incentive mechanism, excluding service-based incentives, to align with the security, fairness, and interoperability requirements of the proposed architecture. The combination of monetary rewards with reputation-gated weighting mitigates Sybil and collusion attacks by imposing economic costs on malicious behavior while preventing wealth dominance through trust-based gating. This approach is compatible with both permissioned and permissionless blockchains, eliminating the necessity for cross-system service standardization. Conversely, service-based incentives require supplementary access-management infrastructure and may create central control points, rendering them less appropriate

for a diverse interoperability environment.

Chapter 3

Related Works

In this section, we begin by categorizing the major design approaches to blockchain interoperability, highlighting the architectural principles and mechanisms that underpin existing solutions. We then provide an overview of established interoperability frameworks, analyzing their core components, design choices, and limitations. Following this, we examine studies that introduce privacy-preserving mechanisms aimed at safeguarding sensitive information during cross-chain interactions. Finally, we review the range of incentive mechanisms proposed in the literature, with a focus on their applicability to decentralized and heterogeneous blockchain environments.

3.1 Blockchain Interoperability

Blockchain interoperability refers to the ability of two or more heterogeneous blockchain platforms—potentially differing in consensus algorithms, governance models, network accessibility, security assumptions, and data structures—to exchange

data, transfer assets, and invoke cross-chain operations. Existing solutions primarily target public blockchains, particularly those used for cryptocurrency exchange. However, limited attention has been given to interoperability between permissioned and permissionless blockchains, where additional challenges such as privacy, anonymity, and incentive alignment must be addressed.

3.1.1 Interoperability between Permissionless/public Blockchains

Polkadot facilitates interoperability among public/permissionless blockchain networks via its *relay chain* architecture, which links customizable *parachains* under a shared security model Wood [2016]. Although effective for connecting permissionless networks, Polkadot's design does not directly address interoperability with permissioned blockchains, whose restricted access policies and privacy requirements demand stricter trust and data-handling guarantees. Furthermore, its decentralized governance and the operational autonomy of parachains can introduce risks such as malicious logic injection or coordinated collusion, making it unsuitable for the regulated and privacy-sensitive environments targeted in our work.

Cosmos implements interoperability through its Inter-Blockchain Communication (IBC) protocol, enabling independent blockchains (*zones*) to exchange data and value via the *Cosmos Hub* Kwon and Buchman [2019]. While the framework can support both permissioned and permissionless systems, its primary focus lies within decentralized finance (DeFi) and public blockchain ecosystems. Incorporating highly regulated or privacy-sensitive permissioned networks remains non-trivial, as compliance constraints, heterogeneous governance models, and stricter data-handling re-

quirements pose significant integration challenges that our proposed architecture is specifically designed to address.

3.1.2 Interoperability Between Permissioned Blockchains

Hyperledger Cacti Foundation [2025] is designed for enterprise-oriented interoperability, facilitating secure communication between permissioned blockchain platforms such as Hyperledger Fabric (HF), Corda Brown et al. [2016], and Quorum Consensys [2023]. Its modular, connector-based architecture enables cross-platform compatibility without requiring modifications to underlying consensus or protocol layers. Nevertheless, Cacti is inherently confined to permissioned-only environments and lacks native support for interoperability with public, permissionless blockchain systems—a limitation that the proposed framework in this thesis explicitly overcomes.

3.1.3 Solutions Bridging Permissioned and Permissionless Blockchains

Axelar Network [2021] provides cross-chain interoperability via its Universal Cross-Chain Communication (UCC) protocol, supported by a decentralized validator network. This design minimizes the need for network-specific modifications, enabling broad compatibility across heterogeneous chains. However, the reliance on an external validator set introduces additional trust assumptions, which can be particularly problematic in high-security or enterprise settings where strict trust boundaries and compliance requirements are paramount.

Chainlink's Cross-Chain Interoperability Protocol (CCIP) Breidenbach et al. [2021] is built on its Decentralized Oracle Network (DON) and offers a gen-

eralized messaging and token-transfer framework across blockchains. CCIP employs off-chain reporting (OCR 2.0) and an anti-fraud network to ensure secure packet delivery between smart contracts on different chains. It abstracts cross-chain communication into a unified layer, enabling developers to build token bridges, programmable Decentralized Applications (dApps), and asset flows without custom bridge code per network.

To address privacy and enterprise compliance requirements, Chainlink introduced CCIP Private Transactions combined with its Blockchain Privacy Manager Breidenbach et al. [2021]. Through end-to-end encryption controlled by institutional clients and selective data exposure policies, institutions can limit the visibility of blockchain node operators into transaction details such as payload or amount. However, reliance on off-chain infrastructure introduces governance and trust dependencies—metadata such as timing and transaction frequency may still leak, and the privacy management remains accessible only to pilot clients. Consequently, CCIP's oracle-based structure may still fall short of strict privacy and decentralization guarantees expected in permissioned blockchain environments.

3.1.4 Privacy-Preserving Techniques in Cross-Chain Systems

Several works have sought to preserve user privacy and ensure secure asset exchange across blockchains. Zamyatin et al. Zamyatin et al. [2019] introduced **Xclaim**, a protocol for cross-chain cryptocurrency-backed asset exchange. While it addresses denial-of-service (DoS) and Sybil attacks, transaction-level privacy remains unaddressed.

Han et al. Han et al. [2025] proposed **P2C2T**, a privacy-focused protocol that emphasizes atomicity, unlinkability, and indistinguishability. Although effective in many security domains, it overlooks collusion and deanonymization threats intrinsic to blockchain architectures, particularly when bridging permissioned and permissionless networks. Most existing privacy-preserving protocols also target atomic swaps, focusing on homogeneous blockchain systems and neglecting network-level security issues that emerge during heterogeneous interoperability.

3.1.5 Incentive Mechanisms in Blockchain Networks

Incentive mechanisms serve two primary roles: (1) attracting a sufficient number of nodes to join the network, thereby ensuring decentralization and avoiding single points of failure, and (2) encouraging nodes to remain active and consistently participate in transaction processing Huang et al. [2019]. A very low number of nodes risks centralization, where a small set of participants could monopolize the system.

Such mechanisms can be realized through various techniques, including game theory, reputation systems, proof-of-work, and proof-of-stake. The effectiveness and security of incentive mechanisms in decentralized environments ultimately depend on participant behavior Li et al. [2020].

Li et al. Li et al. [2023] proposed a game-theory-based incentive mechanism for non-cryptocurrency blockchains, known as GTI. This mechanism adjusts validator payoffs so that the Nash equilibrium aligns with honest, consensus-compliant behavior. Through mathematical analysis and simulation, GTI was shown to promote incentive compatibility under PBFT by rewarding validators who truthfully follow

protocol phases and penalizing non-cooperative actions.

However, GTI's game-theoretic model assumes fully rational players and perfect knowledge of others' actions—conditions that may not hold in real-world or noisy enterprise deployments. Moreover, the mechanism models validator actions as strictly honest or faulty, leaving more subtle forms of misbehavior unaddressed. While GTI leverages payoff adjustments, it lacks concrete economic collateral, which may limit enforceability. Finally, because PBFT scales poorly with increasing node counts, the added overhead of GTI's payoff calculation and enforcement may further hinder performance in larger permissioned networks.

Oliveira et al. Tuler de Oliveira and Others [2020] propose the **Blockchain Reputation-Based Consensus (BRBC)** mechanism as a scalable and resilient solution for distributed, mistrusting applications. In BRBC, only nodes whose reputation exceeds a predefined network trust threshold are eligible to propose blocks. A randomly selected jury of *judge* nodes monitors and evaluates behavior during consensus rounds. Cooperative actions earn reputation-based rewards, while non-cooperative or malicious behaviors trigger punishments or access revocation. The authors provide a security analysis and simulation results showing that BRBC can effectively expel nodes when more than 50% exhibit malicious behavior while maintaining throughput under adversarial conditions.

BRBC, however, employs hard reputation thresholds: nodes falling below the threshold are denied block production rights. This rigid design can penalize nodes suffering temporary performance drops and limit fairness or recovery potential. While reputation governs access, BRBC does not incorporate stake-based collateral, on-

chain rewards, or slashing, reducing its economic deterrence relative to bonded systems.

Vos et al. de Vos et al. [2021] introduce **XChange**, a universal, trust-minimized mechanism for asset exchange between isolated permissioned blockchain platforms. Unlike centralized intermediaries or atomic swaps, XChange decouples trade coordination from asset transfer by recording trade specifications on a separate accounting ledger and enforcing a structured five-phase protocol. This design enables secure asset trading between heterogeneous ledgers while constraining fraud risk, even in resource-constrained environments. Their implementation demonstrates sub-second trade completion with linear scalability and higher throughput compared to decentralized exchanges such as BitShares or Waves.

Nevertheless, XChange depends on trade metadata stored outside the underlying blockchains, which may complicate integration with highly siloed or policy-restricted permissioned systems. Furthermore, XChange does not incorporate explicit stake-based incentives or slashing mechanisms to economically deter malicious behavior. Nodes participate in trade gating but lack token-backed economic guarantees.

3.1.6 Discussion and Research Gap

Blockchain interoperability enables distinct blockchain networks to exchange data and interact seamlessly. Existing interoperability solutions primarily focus on permissionless networks, as these are publicly accessible and inherently decentralized. In contrast, access to permissioned blockchains is restricted by network administrators, preventing unauthorized users from joining or interacting with participants. Con-

sequently, interoperability involving permissioned blockchains has remained largely underexplored.

Nonetheless, participants in permissioned blockchains may still require secure mechanisms for transferring data to external networks while complying with strict privacy and governance policies. To address this gap, we propose a generic interoperability framework that enables connectivity between permissioned and permissionless blockchains without introducing additional assumptions about the underlying networks. Our approach leverages dedicated smart contracts linked to a **bridging component**—a coordinated group of nodes that collaborate to facilitate data transfer. The bridge integrates cryptographic primitives, including ring signatures and blind signatures, to ensure confidentiality, integrity, and unlinkability of transactions. In addition, we design a reputation-based incentive mechanism to reward honest bridge nodes while mitigating risks of collusion Ciccarelli and Cigno [2011] and Sybil attacks.

By integrating privacy-preserving cryptographic techniques with incentive-aligned participation, our framework effectively addresses the crucial challenges of facilitating secure and trustworthy interoperability between permissioned and permissionless blockchains. This is a domain that remains inadequately explored in the current literature.

Chapter 4

System Design

In this chapter, we present our proposed solution along with a novel incentive mechanism for permissioned-to-permissionless cross-communication. We begin by describing the main cryptographic primitives, followed by the anonymization protocols. We then outline the components of our incentive mechanism, which is based on a trust score and designed to remain resilient against Sybil and collusion attacks. Finally, we explain how the trust score is dynamically adjusted to accurately reflect the behavior of participating nodes.

4.1 Cryptographic Primitives

4.1.1 Blind Signatures

Blind signatures Chaum [1983] enable a signer to produce a valid signature on a message without learning the message content. In the canonical RSA construction, a

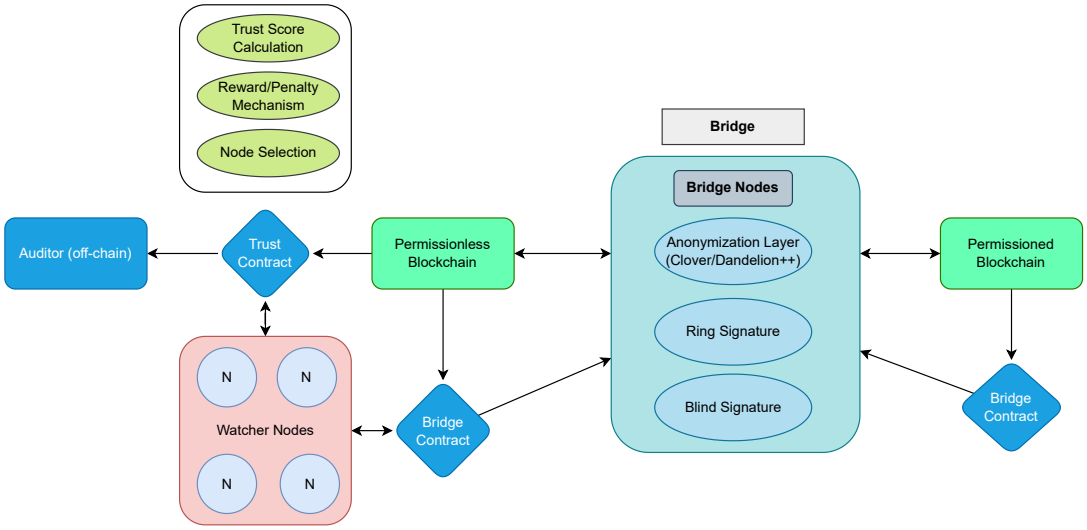

Figure 4.1: Architecture

client chooses a random blinding factor $r \in \mathbb{Z}_N^*$ and computes the blinded message

$$m' = m \cdot r^e \bmod N,$$

where (N, e, d) is the signer's RSA key pair. The signer computes $s' = (m')^d \bmod N$ and returns s' to the client, who unblinds:

$$s = s' \cdot r^{-1} \bmod N.$$

The pair (m, s) is a valid RSA signature on m, yet the signer never observes m in the clear.

In our design, relays obtain blind signatures on *message identifiers* prior to forwarding. When a relay later presents the unblinded signature as proof of authorization, the signer (and any observer of the issuance interaction) cannot directly link the issuance event to the later presentation. This unlinkability reduces the risk that the signer or an auditing service will deanonymize which relay forwarded which message.

4.1.2 Ring Signatures

Ring signatures Rivest et al. [2001] provide signer anonymity within a set (ring) of public keys. A verifier can confirm that a signature was produced by some member of the ring but cannot determine which member. Suppose the ring is $\mathcal{R} = \{PK_1, \ldots, PK_n\}$ and a relay holding SK_i signs message m. The signature $\sigma = (c_1, s_1, \ldots, s_n)$ is computed by choosing random s_j for $j \neq i$ and solving a cyclic relation:

$$c_{j+1} = H\big(m,\, g^{s_j} \cdot PK_j^{c_j}\big), \quad j = 1, \ldots, n, \quad c_{n+1} = c_1.$$

Here, c_1 is the initial challenge value included in the signature. During verification, the verifier reconstructs the sequence $(c'_1, c'_2, \ldots, c'_{n+1})$ using the published $(s_1, \ldots, s_n)$ and the same recurrence:

$$c'_{j+1} = H\big(m,\, g^{s_j} \cdot PK_j^{c'_j}\big), \quad j = 1, \ldots, n.$$

The signature is valid if the reconstructed value wraps around consistently ($c'_{n+1} = c_1$). This condition ensures that the cyclic relation holds, proving that one (but not which) member of the ring signed the message.

In our bridge, ring signatures conceal the identity of the specific relay within a quorum that submitted or endorsed a packet, thus ensuring sender anonymity among the participating relays.

4.1.3 Soft-Consensus Quorum Gate

Our *soft consensus* mechanism requires at least t out of n relays to submit *identical* payload hashes before the Bridge Contract finalizes a cross-chain message. Formally,

let $\{h_i\}_{i \in R}$ denote the multiset of submitted hashes. The contract accepts if

$$\exists h^* : \quad \left|\{\, i \in R : h_i = h^* \}\right| \;\geq\; t.$$

This threshold-based mechanism is lightweight and ensures correctness in the presence of up to $n - t$ faulty or malicious relays.

4.2 Incentive Mechanism

We propose an incentive mechanism that rewards participating nodes based on their observed behavior, which directly influences the overall performance of the bridging component. To capture this, we define a **trust score** computed from three key metrics: **participation rate**, **success rate**, and **average forwarding delay**.

The design goals of our incentive mechanism are:

1. Capturing recent measurable performance while retaining long–term reputation.
2. Remaining robust against Sybil and stake-grinding attacks.
3. Enabling early detection and penalization of collusion.
4. Being gas-efficient when implemented on a permissionless smart-contract platform.

4.2.1 Per-epoch Observable Metrics

The Trust Contract updates relay trust scores based on concrete, verifiable evidence collected in each epoch. Three categories of observable metrics serve as inputs:

1. **On-chain anchor events:** bridge receipts and confirmations that a message has been successfully finalized on the destination chain.

2. **Relay-generated proofs:** cryptographic signatures that attest to correct message forwarding during the relay process.

3. **Watcher attestations:** signed statements from independent monitoring entities that either confirm relay honesty or challenge observed misbehavior.

Together, these inputs ensure that trust score updates are evidence-driven, transparent, and resistant to manipulation by any single actor.

On-chain computation of lightweight metrics Metrics such as participation rate $p_i^{(t)}$, success rate $s_i^{(t)}$, delay score $d_i^{(t)}$, and validated challenge rate $c_i^{(t)}$ are computed directly from these events:

- $p_i^{(t)}$ is the ratio between the number of valid proofs submitted by relay i and the total number of forwarding opportunities in epoch t.

- $s_i^{(t)}$ is the fraction of i's submissions whose proofs both match a corresponding bridge receipt and pass signature verification.

- $d_i^{(t)}$ is derived from the average forwarding delay $\bar{d}_i^{(t)} = \text{avg}(t_{\text{receipt}} - t_{\text{send}})$, with the exponential transform $d_i^{(t)} = e^{-\lambda \bar{d}_i^{(t)}}$ applied to bound scores in $(0, 1]$ and reward lower latency.

- $c_i^{(t)}$ is computed as the fraction of i's submissions that are successfully challenged and verified as incorrect by the Trust Contract.

Off-chain computation of expensive metrics: The collusion indicator $\kappa_i^{(t)}$ requires computing pairwise correlation statistics between relays to detect unusually high agreement within a small subset. Such computations are resource-intensive and not practical on-chain. Instead, a *trusted off-chain auditor* retrieves the necessary event data from the blockchain, computes $\kappa_i^{(t)}$ for each relay, and commits the results to the blockchain in a compact form. This is achieved by constructing a Merkle tree over all κ values, publishing the Merkle root along with the auditor's digital signature, and having the Trust Contract verify the signature before accepting the values.

This hybrid on-chain/off-chain approach ensures that routine checks are executed deterministically on-chain for transparency, while computationally heavy analyses are performed off-chain to maintain efficiency. The cryptographic anchoring of off-chain results back onto the blockchain preserves verifiability without incurring excessive on-chain costs, enabling the system to scale without sacrificing trust guarantees.

4.2.2 Per-epoch performance score

The individual performance metrics defined in the previous section are combined into a single bounded value to represent a relay's performance within a given epoch. Let $w_p, w_s, w_d, w_c, w_\kappa \geq 0$ be nonnegative weights representing the relative importance assigned to participation rate, success rate, delay score, validated challenge rate, and collusion indicator, respectively. The first three weights satisfy $w_p + w_s + w_d = 1$, ensuring that the positive contributions are normalized. The per-epoch performance score for relay i during epoch t is defined as:

$$R_i^{(t)} \;=\; w_p\, p_i^{(t)} \;+\; w_s\, s_i^{(t)} \;+\; w_d\, d_i^{(t)} \;-\; w_c\, c_i^{(t)} \;-\; w_\kappa\, \kappa_i^{(t)}. \tag{4.1}$$

Positive terms increase the score in proportion to good performance (high participation, success, and low delay), while negative terms decrease the score in proportion to undesirable behavior (high challenge rate or suspected collusion). By construction, $R_i^{(t)}$ is bounded between $-(w_c + w_\kappa)$ and 1, with the lower bound representing the worst possible behavior and the upper bound corresponding to ideal performance.

To map this range into a standardized $[0, 1]$ interval, we apply a rescaling function:

$$\sigma(R) \;=\; \frac{R + w_c + w_\kappa}{1 + w_c + w_\kappa}, \tag{4.2}$$

such that $R = -w_c - w_\kappa \mapsto 0$ and $R = 1 \mapsto 1$. This normalization enables a consistent interpretation of the score across deployments with varying weight configurations.

4.2.3 Exponential moving average trust update

Let $T_i^{(t)} \in [0, 1]$ denote the trust score of relay i at the beginning of epoch t. To ensure that the trust score reflects both recent performance and historical reliability, we update it using an exponentially weighted moving average (EWMA) Hunter [1986]:

$$T_i^{(t+1)} \;=\; \mathrm{clip}_{[0,1]}\Big(\alpha\, T_i^{(t)} + (1-\alpha)\,\sigma\big(R_i^{(t)}\big)\Big), \tag{4.3}$$

where $\alpha \in [0, 1)$ controls the balance between historical memory and responsiveness to new performance measurements. Larger values of α emphasize long-term stability, while smaller values allow the score to react more quickly to recent changes.

The clipping function $\mathrm{clip}_{[0,1]}$ ensures that the trust score remains within the valid range, regardless of extreme fluctuations in performance. By design, this EWMA-based update mechanism smooths short-term noise while enabling sustained positive or negative behavior to significantly influence the trust score over time.

4.2.4 Stake and Effective Weight Calculation

To resist Sybil attacks and enforce meaningful slashing, the architecture combines each relay's normalized stake with its reputation score to compute an *effective weight* $w_i^{(t)}$. This weight is subsequently used for both quorum decisions and proportional reward allocation. Slashing refers to the punitive reduction of a relay's staked assets when it engages in malicious or faulty behavior, thereby creating a strong economic deterrent against misbehavior.

Stake normalization The raw on-chain collateral S_i deposited by relay i is first normalized via a concave logarithmic transformation:

$$\widetilde{S}_i = \frac{\log(1 + S_i)}{\log(1 + S_{\max})} \in [0, 1], \tag{4.4}$$

where $S_{\max}$ is a fixed system parameter representing the upper bound for normalization. The logarithmic form ensures diminishing returns on additional stake, limiting wealth concentration.

Reputation gating Stake influence is gated by the relay's trust score $T_i^{(t)}$ using:

$$w_i^{(t)} = T_i^{(t)} \cdot \big(\omega_R + (1 - \omega_R)\, \widetilde{S}_i\big), \tag{4.5}$$

A low ω_R undervalues the importance of reputation and emphasizes more on the stake. As a result, a group of malicious nodes can buy or split stake into many sybil identities and gain disproportionate influence, even with no reputation history. On the other hand, a high ω_R makes reputation a dominating factor. This makes it difficult for new but honest relays (who start with no reputation) to gain meaningful influence,

even if they put up significant stake. This ultimately discourages participation and slows onboarding.

4.2.5 Weighted Quorum and Reward Distribution

During cross-chain message finalization, the Bridge Contract applies an *effective-weight quorum rule*. A candidate message hash h^* is accepted if the total effective weight of relays endorsing h^* meets or exceeds a threshold fraction τ of the total system weight:

$$\sum_{i:\, h_i = h^*} w_i^{(t)} \geq \tau \cdot \sum_k w_k^{(t)}, \tag{4.6}$$

At a very low τ, the system can finalize a transaction with only a bare majority. That allows adversaries controlling just over 50% of the weight to succeed, which is unsafe. A high τ, however, forces the system to require near-unanimous agreement. That ensures strong safety, but in practice some honest nodes may be offline or delayed, which damages liveness of the system.

At the end of each epoch, the reward pool $R_{\text{pool}}^{(t)}$ is distributed proportionally to effective weights:

$$\text{Reward}_i^{(t)} = R_{\text{pool}}^{(t)} \cdot \frac{w_i^{(t)}}{\sum_k w_k^{(t)} + \varepsilon}, \tag{4.7}$$

with a small constant $\varepsilon > 0$ to prevent division-by-zero edge cases. This proportional allocation incentivizes both high stake and sustained good performance.

4.2.6 Collusion Detection via Agreement Statistics

A collusion indicator $\kappa_i^{(t)}$ is computed for each relay. For every message index m in epoch t, the binary indicator $1_{i,j}^{(m)}$ denotes whether relays i and j submitted identical

candidate hashes. The *weighted agreement score* for relay i is:

$$A_i^{(t)} = \frac{\sum_{j \neq i} w_j^{(t)} \sum_m 1_{i,j}^{(m)}}{\left(\sum_{j \neq i} w_j^{(t)}\right) N^{(t)}}, \tag{4.8}$$

where $N^{(t)}$ is the number of candidate messages in that epoch. This score measures how frequently i aligns with others, weighted by their effective weights.

The system-wide weighted mean $\mu_A^{(t)}$ and standard deviation $\sigma_A^{(t)}$ are computed over all $A_i^{(t)}$ values. Relay i's collusion indicator is then:

$$\kappa_i^{(t)} = \text{clip}_{[0,1]}\left(\frac{A_i^{(t)} - \mu_A^{(t)}}{\kappa_\sigma \, \sigma_A^{(t)}}\right), \tag{4.9}$$

where $\kappa_\sigma > 0$ sets the sensitivity to anomalous agreement. High $\kappa_i^{(t)}$ values reduce the performance score $R_i^{(t)}$ in Eq. (4.1), thereby penalizing collusion tendencies in subsequent trust updates.

4.3 System Components: Overview

The architecture introduces the following components to establish interoperability, as illustrated in Figure 4.1:

1. **Watcher Nodes (permissioned observers):** A cohort of nodes (one or more) that monitor the permissioned chain for anchor events and produce signed attestations. Watchers are protocol participants: they submit attestations to the Bridge Contract and are economically incentivized for correct reporting. Watchers form the first line of independent verification of source events.

2. **Relay Cohort (forwarders / bridge service):** The set of semi-trusted service nodes that implement the bridge's forwarding logic: they receive candidate

messages (from watchers or the anchor interface), propagate them through the relay network according to the chosen anonymization policy, sign/endorse relay receipts, and submit candidate hashes to the Bridge Contract. Operationally, the relay cohort implements forwarding, proof-generation, and interaction with on-chain contracts.

3. **Bridge Contract (collector / finalizer):** The on-chain smart contract deployed on the destination (permissionless) chain that collects relay submissions (candidate hashes and accompanying attestations), applies the quorum (weighted) acceptance rule, finalizes accepted messages, and emits receipts that downstream applications and the Trust Contract consume. The Quorum Gate logic is implemented here.

4. **Trust Contract (reputation & reward manager):** A permissionless-chain smart contract that stores per-relay metadata (trust scores, stake commitments), ingests compact evidence (watcher attestations, auditor summaries), computes epochal trust updates, and executes reward/penalty distributions. The Trust Contract produces *effective weights* that the Bridge Contract uses for quorum decisions.

5. **Off-chain Auditor (auxiliary aggregator):** An independent off-chain service that gathers detailed logs and computes expensive analytics (pairwise agreement, collusion indicators, statistical summaries). The auditor publishes compact, signed summaries (e.g., Merkle roots) to the Trust Contract for verification. The auditor is auxiliary and its reports are treated as evidence, not as

unilateral authority.

4.4 Cross-Chain Transaction Life-cycle

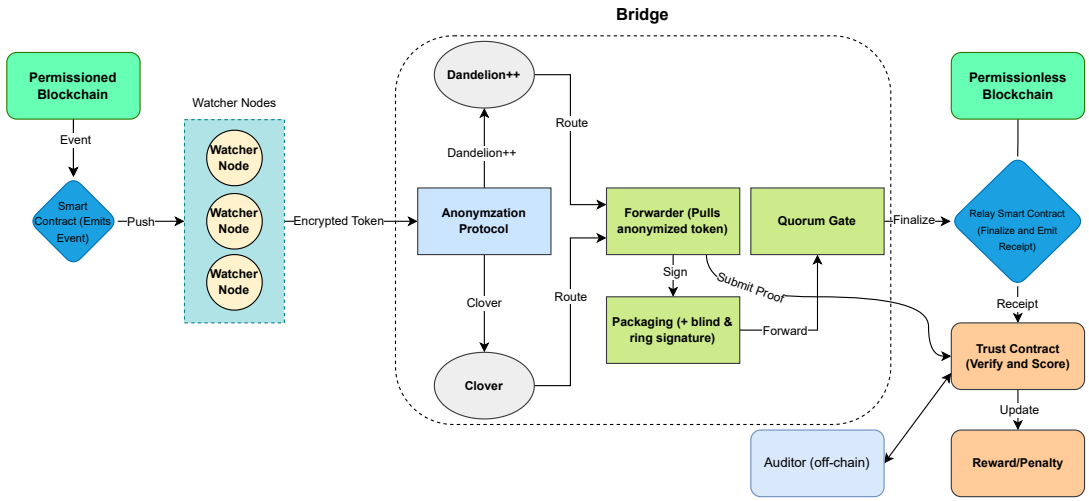

Figure 4.2: System Transaction Lifecycle

This section details the end-to-end transaction flow from the permissioned blockchain to the permissionless blockchain, highlighting all intermediate cryptographic and protocol steps.

1. **Transaction Creation and Event Emission on permissioned network**

   A client submits a transaction m to the permissioned network, invoking smart contract logic. Upon successful endorsement and ordering, the smart contract

emits an event $\mathsf{Event}(h, t_f)$, where

$$h = \mathsf{Hash}(m \,\|\, \mathsf{nonce}), \tag{4.10}$$

$$t_f = \text{permissioned network block timestamp.} \tag{4.11}$$

This event serves as the trigger for cross-chain forwarding.

2. **Observation by Watcher Nodes**

   A set of n_w *watcher* nodes subscribe to permissioned network events. Upon seeing $\mathsf{Event}(h, t_f)$, each watcher w_i computes an *encrypted token*

$$\mathsf{token}_i = \mathsf{Enc}_K\big(h \,\|\, t_f \,\|\, r_i\big),$$

   where r_i is a fresh random nonce and K is a symmetric group key. All tokens are indistinguishable and unlinkable.

3. **Network-Level Obfuscation**

   Each watcher pushes token_i into the chosen anonymization protocol:

   - Clover, which forwards tokens along a random walk of length L before broadcast.
   - Dandelion++, which runs a stem phase of length k followed by a fluff (broadcast) phase.

   This step hides IP and timing metadata from external observers.

4. **Token Retrieval by Forwarder Nodes**

   A forwarder node f_j obtains one of the anonymized tokens, denoted token, and

then decrypts it to proceed with the forwarding process.

$$(h, t_f, r) = \mathsf{Dec}_K(\mathsf{token}),$$

and uses h to fetch or reconstruct the full payload m from a secure off-chain store.

5. **Blind and Ring Signature Generation**

   The forwarder obtains a *blind signature* on h (preventing linkage to its identity) via $\mathsf{BSign}(h)$ and then computes a *ring signature* σ over the tuple (h, t_f) using the ring of current relay public keys $\mathcal{R}$. The resulting packet is:

$$P = \big(h,\, t_f,\, \mathsf{BSign}(h),\, \sigma\big).$$

6. **Soft-Consensus Quorum Submission**

   Forwarders submit P to the Bridge Contract on the permissionless blockchain. Let T be the multiset of reported hashes. The contract finalizes when

$$\exists\, h^* : \big|\{\, P_i : h_i = h^* \}\big| \geq t,$$

   for threshold $t \leq n_f$, forming a t-of-n_f quorum agreement.

7. **Bridge Finalization and Receipt Emission**

   Upon reaching quorum, the Bridge Contract executes the cross-chain logic and emits a receipt event $\mathsf{Receipt}(h^*, t_s)$, where t_s is the permissionless blockchain block timestamp.

8. **Proof Submission to Trust Contract**

   Each forwarder f_j constructs a proof (h^*, t_f, t_s) and invokes

`submitProof`$(h^*, t_f, t_s, \mathsf{BSign}(h^*), \sigma)$ on the Trust Contract. The contract verifies:

$$t_s - t_f \leq D_{\max}, \quad \mathsf{VerifyBlind}(\mathsf{BSign}(h^*)), \quad \mathsf{VerifyRing}(\sigma),$$

ensuring correctness and timeliness.

9. **Trust Score Update and Reward/Penalty**

   The Trust Contract updates each f_j's score T_j:

   $$T_j \leftarrow \alpha T_j + \beta + \gamma e^{-\lambda(t_s - t_f)}.$$

   Rewards are distributed proportionally, and penalties (score reduction or stake slashing) apply for failures or late proofs.

10. **(Optional) Reverse-Flow Response**

    If bidirectional interaction is required, the Bridge Contract may emit an event on the permissionless blockchain that watcher/forwarder nodes observe and relay back to the permissioned network via analogous steps.

Chapter 5

Security Analysis

In this chapter, we analyze the security guarantees of the proposed architecture, with particular emphasis on its resilience against privacy and incentive-related attacks. We begin by applying the STRIDE threat model to systematically examine the foundational security concerns and describe how each threat is mitigated in our design. We then provide a mathematical justification for the robustness of our incentive mechanism, demonstrating its resistance to Sybil and collusion attacks.

5.1 Threat Model

To evaluate the privacy and security of our system, we adopt the STRIDE threat model Shostack [2014], which classifies six primary categories of security threats in software systems: spoofing, tampering, repudiation, information disclosure, denial of service, and elevation of privilege. While STRIDE provides a systematic framework for identifying foundational security risks, decentralized environments also introduce

the additional challenge of collusion attacks. We first analyze our system under the STRIDE categories and subsequently evaluate its robustness against collusion.

5.1.1 Spoofing

Spoofing threats involve an adversary attempting to impersonate a legitimate relay node in order to gain unauthorized access Shostack [2014]. In our system, each relay must register an identity anchored by a cryptographic key-pair and collateral stake. All submitted proofs and signatures are verifiable on-chain by the Bridge and Trust Contracts. Since participation is gated by both key ownership and reputation, adversaries cannot successfully impersonate relays without the corresponding private keys and stake deposits. Spoofing is therefore effectively mitigated.

5.1.2 Tampering

Tampering refers to unauthorized modification of data in transit or storage Shostack [2014]. In our architecture, message receipts and proofs are cryptographically signed by the source chain and verified by the Bridge Contract before acceptance. Any relay attempting to alter the message or proof will fail signature verification. Furthermore, watcher nodes are empowered to challenge invalid submissions, ensuring that tampering attempts are detected and penalized via slashing.

5.1.3 Repudiation

Repudiation involves denying the performance of an action when responsible Shostack [2014]. In our system, all relay submissions, challenges, and adjudications

are permanently recorded as verifiable on-chain events. Each relay's actions contribute directly to its participation and success metrics, which are publicly auditable. As a result, relays cannot plausibly deny having submitted or failed to submit a message.

5.1.4 Information Disclosure

Information disclosure pertains to unauthorized leakage of sensitive data Shostack [2014]. In bridging between permissioned and permissionless networks, sender anonymity and confidentiality are critical. To address this, the system employs ring signatures, blind signatures, and source-anonymization protocols. Relays and watchers only observe blinded or encrypted payloads, preventing them from inferring the identity of the sender or the contents of confidential transactions.

5.1.5 Denial of Service (DoS)

DoS attacks aim to disrupt system availability by preventing honest relays from participating or by flooding the bridge Shostack [2014]. The quorum mechanism ensures liveness so long as a sufficient fraction of effective weight behaves honestly, while the stake requirement discourages large-scale Sybil-based spam. Additionally, the trust update mechanism penalizes relays that repeatedly fail to participate or that introduce excessive delay. These measures mitigate protocol-level DoS threats. However, network-level flooding attacks remain outside the scope of the proposed architecture. Mitigating DoS attacks becomes more challenging especially when anonymization protocols such as Dandelion++ are adopted Times [2023].

5.1.6 Elevation of Privilege

Elevation of privilege occurs when an adversary gains unauthorized control or disproportionate influence Shostack [2014]. In our design, effective weight is determined by the product of trust score and concavely normalized stake, which prevents actors from accumulating power solely through financial resources. Moreover, the collusion indicator penalizes suspiciously coordinated behavior, and slashing mechanisms deter malicious influence attempts. Consequently, elevation-of-privilege threats are strongly mitigated.

5.2 Sybil, 51%, and Collusion Attacks

5.2.1 Sybil resistance

A fundamental requirement for the security of a cross-chain relay system is resistance to Sybil attacks, where an adversary spawns many identities to amplify their influence at negligible cost. In our design, this threat is addressed through a combination of stake-based admission control and reputation-gated weighting. First, every relay must provide a minimum collateral deposit $S_{\min} > 0$ to register, thereby imposing an irreducible cost on each additional identity. Second, stake is normalized via a concave function (Eq. (4.4)), which ensures that marginal returns diminish rapidly with increasing stake. This prevents wealth concentration from overwhelming the system while still discouraging Sybil inflation. Finally, effective weight is multiplicatively gated by trust score $T_i^{(t)}$ (Eq. (4.5)), so newly-created identities, even if staked, contribute negligible influence until they establish a history of correct participation.

The combined effect is that Sybil attacks require both sustained honest behavior and collateral commitment, thereby raising their economic cost and limiting their feasibility.

5.2.2 Weighted-majority (51%) mitigation

In decentralized systems, a natural concern is whether an adversary who attracts a large fraction of effective weight can finalize malicious candidates under the quorum rule. In our protocol, the Bridge Contract accepts a candidate message only if the aggregated effective weight supporting it exceeds a threshold fraction $\tau \in [0.66, 0.75]$ of the total (Eq. (4.6)). Thus, an adversary must acquire at least $\tau \cdot W_{\text{total}}^{(t)}$ effective weight in order to compromise finality. Since effective weight depends jointly on stake and reputation, the adversary must not only lock up substantial capital but also sustain honest performance over many epochs to build sufficient trust. This coupling of economic and temporal cost significantly increases the difficulty of a 51% attack (weighted-majority attack). Compared to pure stake-based protocols, which allow sudden takeovers via large deposits, or pure reputation-based systems, which are vulnerable to long-term infiltration, our hybrid scheme ensures that adversarial majorities are both expensive to obtain and slow to build, providing the Bridge with a strong safety margin.

5.2.3 Collusion detection and response

Even if Sybil inflation and majority control are infeasible, adversaries may attempt to collude by coordinating submissions among a small group of relays. Such behavior

can bias quorum formation or suppress diversity of opinions. To mitigate this, we introduce a collusion indicator $\kappa_i^{(t)}$ (Eq. (4.9)), which measures whether a relay's agreement rate with a small subset of peers is statistically anomalous relative to the population. Relays with unusually high pairwise agreement scores are flagged as potential colluders. Upon detection, the Trust Contract enforces defensive measures: (i) challenge rewards are dynamically increased to incentivize external verification in the suspicious epoch, (ii) the trust scores T_i of implicated relays are reduced via the exponential moving average update (Eq. (4.3)), and (iii) the Bridge Contract can require additional watcher attestations before finalizing any message supported primarily by the colluding set. These responses jointly reduce the effective weight of colluders, expose them to greater challenge risk, and limit their capacity to bias quorum outcomes.

5.2.4 Informal correctness claim

Claim If an adversary controls strictly less than a τ-fraction of the total effective weight, then the Bridge cannot finalize a malicious candidate under the quorum rule (Eq. (4.6)). Furthermore, the combination of multiplicative reputation gating and collusion penalties further ensures that attempts to inflate influence via rapid staking or clique formation are both economically costly and probabilistically detectable.

Proof By the quorum condition, the Bridge finalizes a candidate only if the supporting effective weight exceeds $\tau \cdot W_{\text{total}}^{(t)}$. If the adversary controls strictly less than this threshold, then their candidate cannot be accepted by the quorum condition. To cross the threshold, the adversary must accumulate additional effective weight, which

requires both stake and reputation (Eq. (4.5)). Reputation is incrementally earned through correct participation and is reduced under collusion detection (κ). Thus, unlike pure stake-based protocols, the adversary cannot acquire decisive influence through a sudden capital injection; unlike pure reputation-based systems, a colluding clique cannot sustain anomalous agreement without triggering penalties. Instead, they face escalating costs in the form of locked stake, slashing exposure, and diminishing trust under anomalous behaviour. Hence, the attack surface is limited, and the adversary's success probability remains negligible unless they cross the τ-fraction threshold.

5.3 Source Anonymization Security Analysis

To mitigate *de-anonymization attacks*, our system leverages anonymization protocols that obfuscate the origin of transactions by routing messages through randomly selected relays in a forwarding path. Such attacks, as described in Zhang et al. [2020], exploit the susceptibility of the initial broadcasting node, enabling adversaries to infer transaction origins if enough forwarding relays are compromised.

5.3.1 Adversarial Model

Let the forwarding path consist of f relays, drawn from a set of n total relays. In our design, relays are not chosen uniformly at random but are instead selected with probability proportional to their trust-weighted score $w_i^{(t)}$ (Eq. 4.5). Thus, the

effective probability of a malicious relay being chosen is

$$p_{\mathsf{adv}} = \sum_{i \in \mathcal{A}} \frac{w_i^{(t)}}{\sum_{k=1}^{n} w_k^{(t)}},$$

where $\mathcal{A}$ denotes the set of relays controlled by adversaries. In contrast to uniform selection, this approach biases exposure towards high-trust relays, which are limited in quantity yet strategically appealing targets.

5.3.2 Path Compromise Probability

A transaction is deanonymized when an adversary associates the message's origin with the initial legitimate relay in the pathway. This necessitates the adversary to infiltrate all relays leading to that legitimate node. Assuming independence, the likelihood that the initial j relays are adversarial is

$$p_{\mathsf{deanon}}(j) = (p_{\mathsf{adv}})^j \cdot (1 - p_{\mathsf{adv}}).$$

Aggregating over all possible positions of the first honest relay yields the overall deanonymization probability for a path length f:

$$P_{\mathsf{deanon}}(f) = \sum_{j=0}^{f-1} (p_{\mathsf{adv}})^j \cdot (1 - p_{\mathsf{adv}}).$$

This expression captures the trade-off between f (longer paths increase anonymity but also adversarial exposure) and p_{adv} (adversarial penetration weighted by trust scores).

5.3.3 Effect of Local Visibility

Similar to Dandelion++, each relay exclusively monitors its predecessor, itself, and its successor. Consequently, a singular compromised relay endangers a maximum

of three nodes inside the pathway. Complete path reconstruction necessitates the attacker to infiltrate alternating relays, totaling a minimum of $f/3$ nodes. Consequently, the realistic adversarial endeavor scales linearly with f, rendering large-scale deanonymization expensive.

5.3.4 Watcher and Auditor Oversight

In contrast to conventional P2P networks, our architecture incorporates watcher nodes and an off-chain auditor. These entities can identify the clustering of hostile relays or anomalous correlation patterns in path selection. The effective p_{adv} is constrained by both relay compromise and the likelihood of evading discovery.

5.3.5 Takeaways

From this analysis, we conclude:

- Longer forwarding paths (f) increase baseline anonymity but raise exposure to compromised relays.
- A larger relay set (n) and lower adversarial penetration reduce p_{adv} and thus P_{deanon}.
- Trust-weighted selection makes targeting high-trust relays more attractive, shifting the adversary's strategy from quantity to quality of infiltration.
- Watcher and auditor oversight further lower the effective breach probability by detecting malicious relay behavior.

Consequently, anonymity within our system is probabilistically preserved: adversaries encounter escalating challenges as n increases and supervision mechanisms are included, although the risk of deanonymization cannot be entirely eradicated.

Chapter 6

Evaluation

This chapter evaluates the proposed architecture with respect to our research questions **RQ1**–**RQ3**. First, to address **RQ1** (secure interoperability with privacy preservation), we measure end-to-end latency, throughput, and communication overhead under varying anonymization settings (ring size, stem/path length), and quantify sender-anonymity via deanonymization success rates. Second, to address **RQ2** (security and privacy threats), we examined the system against Sybil, majority-weight, and collusion adversaries, reporting attacker success probabilities, robustness under parameter sweeps (e.g., quorum threshold τ), and the impact of the collusion indicator on misbehaving cohorts. Finally, to address **RQ3** (incentivizing honest behavior), we evaluate the reputation–stake mechanism by tracking trust scores over epochs, reward dispersion (fairness) across relay populations, and the effect of penalties on strategic deviations.

To that end, we first summarize the implementation configuration (relay cohort size, contract parameters, and anonymization settings), then present results along

the three dimensions above: (i) interoperability performance and privacy guarantees (**RQ1**), (ii) security resilience against modeled adversaries (**RQ2**), and (iii) incentive outcomes and equilibrium behavior under the proposed trust model (**RQ3**). The combined evidence demonstrates that the design achieves low overhead while preserving sender anonymity, resists realistic attack models, and aligns individual incentives with protocol correctness.

6.1 Prototype and Evaluation Environment

To validate the feasibility of our design, we have implemented a prototype of the proposed interoperability system.

6.1.1 Programming Environment

The smart contracts were implemented in `Solidity` (version 0.8.x) and deployed to a local Ethereum-compatible testnet using `Hardhat`. The bridge infrastructure, including relays and watcher nodes, was implemented in `Python 3.10` with cryptographic primitives provided by the `PyCryptodome` library. For verifiable randomness, we integrated the VRF library available in the Substrate development framework.

6.1.2 Simulation Setup

To approximate a realistic deployment, we instantiated $n = 20$ relay nodes and $m = 5$ watcher nodes within a `Docker` network. Nodes communicated via `gRPC` channels, and transaction traffic was generated according to a Poisson process with

mean arrival rate $\lambda = 10$ messages per second. The auditor was instantiated as a separate service, written in `Rust`, which aggregated relay and watcher reports every epoch (set to 60 seconds). The code is available on GitHub [1].

6.1.3 Evaluation Metrics

Performance metrics such as message delivery latency, throughput, and trust score evolution were measured using Python-based logging scripts. On-chain gas consumption was measured directly from the Hardhat environment. All experiments were executed on a workstation with an Intel i7-12700 CPU, 16 GB RAM, and Ubuntu 22.04.

6.2 Evaluation Results

This section presents the results of our experimental evaluation, focusing on both incentive alignment and security properties of the proposed system. We analyze fairness in reward distribution, the performance impact of anonymity mechanisms, and the parameter-dependent safety–liveness tradeoff. In addition, we evaluate the evolution of trust scores, the distribution of rewards, and the resilience of the system against adversarial stake concentration and deanonymization attempts. Each subsection introduces the corresponding figure, explains what it shows, interprets the results, and highlights the design implications.

[1] `https://github.com/tcdt-lab/SubstrateFabricIntegration`

6.2.1 Figure 6.1: Reward inequality over time (Gini coefficient)

What the figure shows: This plot displays the epoch-by-epoch Gini coefficient computed from per-epoch reward allocations. The curve rises sharply during the initial reputation-formation phase and then stabilizes with modest fluctuations.

Interpretation: The initial rise is attributed to stochastic early advantages that become amplified through the EWMA reputation update alongside the multiplicative weighting of stake and reputation. Once the EWMA settles, the Gini stabilizes: long-term concentration is modest because reputation dynamics and enforced challenge penalties prevent runaway domination.

Implications: A large α results in slower changes and longer-lived inequality, while a smaller α reduces long-term inequality but may increase volatility. This temporal profile indicates that α must be chosen to balance stability and responsiveness. If necessary, we should consider implementing onboarding protection (temporary weight boosts) to mitigate early inequality.

6.2.2 Figure 6.2: Latency CDF (anonymity vs. performance trade-off)

What the figure shows: Finalization latency empirical CDF for multiple configurations (baseline, Clover, Dandelion++, plus cryptographic overheads for ring/blind signatures). Although the curves are closely clustered, slight changes in median and tail latencies are observed due to anonymization and cryptographic processing.

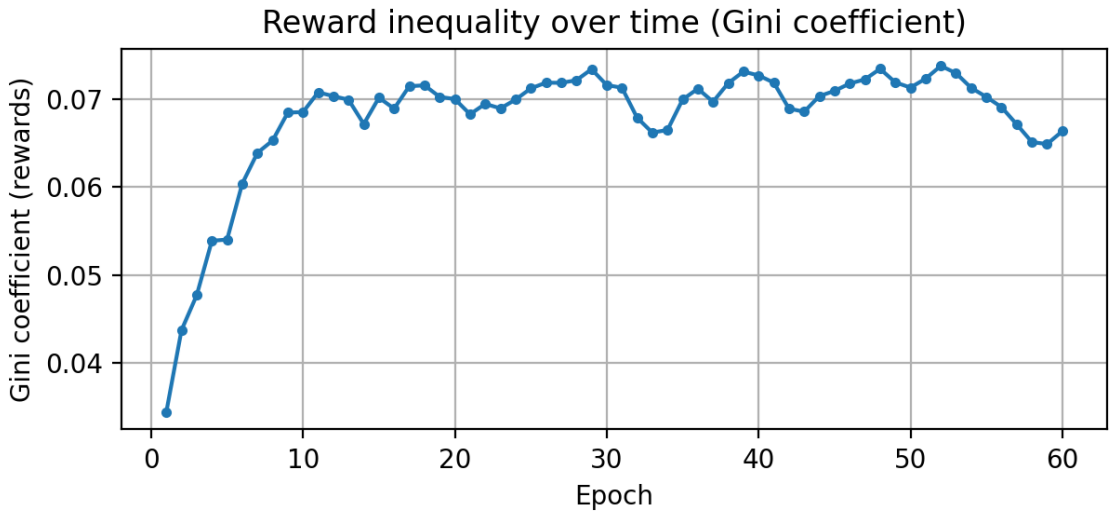

Figure 6.1: Reward inequality over time, measured using the Gini coefficient per epoch. The Gini rises during the initial reputation-formation phase and stabilizes thereafter, indicating a fair but incentive-aligned distribution in the steady state.

Interpretation: Through additional hops, dummy traffic, and processing, anonymity mechanisms raise the latency of each message. The strongest effect appears in the tail of the distribution (higher percentiles), where additional hops and sporadic drops by colluders result in longer completion times; median latency increases only marginally.

Implications: To meet stringent low-latency requirements in application service-level agreements (SLAs), it is essential to adjust anonymization parameters such as path lengths and stem lengths, or to apply anonymization selectively to messages that are privacy-sensitive. Additionally, the implementation of retries and timeouts is crucial while monitoring tail latency as a key metric.

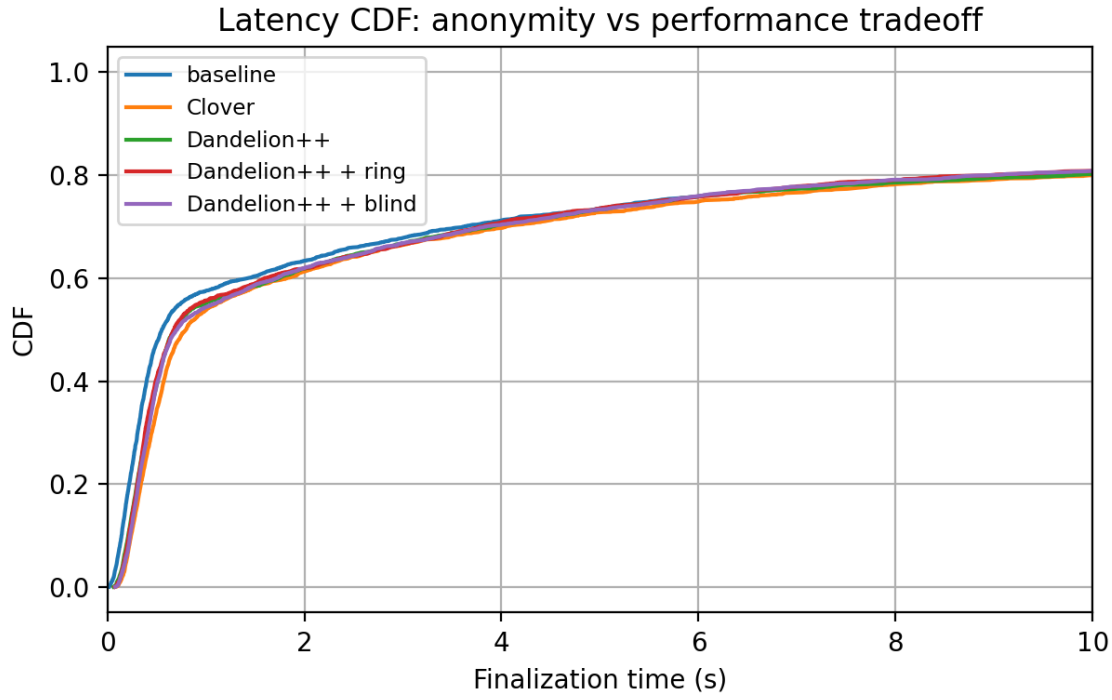

Figure 6.2: CDF of end-to-end message finalization latency for different anonymization protocols and cryptographic configurations. The anonymization mechanisms (Clover, Dandelion++) and cryptographic processing (ring and blind signatures) introduce modest latency shifts relative to the baseline.

6.2.3 Figure 6.3: Parameter sweep — quorum threshold τ vs. EWMA memory α

What the figure shows: Two heatmaps over the (α, τ) grid: (left) estimated adversary success probability under the experiment's proxy adversary model; (right) median finalization latency. In the run depicted, adversary success remained low across the sampled grid, whereas latency exhibited significant variation throughout the grid.

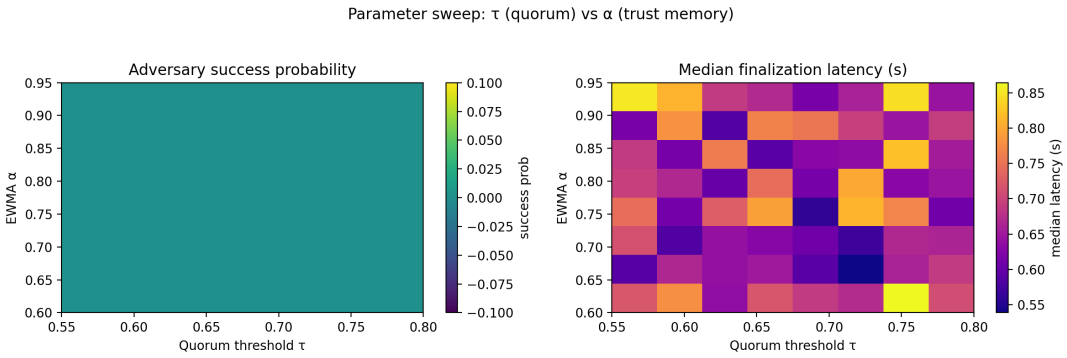

Figure 6.3: Parameter sweep results for quorum threshold τ and EWMA memory α. Left: estimated adversary success probability under the tested model. Right: median finalization latency. The results suggest a tunable safety–liveness tradeoff and indicate a viable operating region with low adversary success and acceptable latency.

Interpretation: The findings demonstrate the anticipated safety–liveness tradeoff: an increase in τ enhances the threshold for adversary success while generally leading to increased latency due to the necessity of accumulating a greater effective weight prior to message finalization. The parameter α influences the responsiveness of reputation; an elevated α results in slower changes to reputation, potentially increasing latency during periods when high-weight relays are temporarily inaccessible.

Implications: In conservative deployments, τ should be selected close to the upper limit of the suggested range and α within a moderate interval (e.g., $\alpha \in [0.75, 0.9]$). To address critical latency, one should decrease τ or shorten the anonymization path length, while being aware of the associated shift in risk.

6.2.4 Figure 6.4: Trust score time series

What the figure shows: The plot illustrates sample trust-score trajectories for a limited number of relays throughout the entire simulation, including several representative honest nodes and a few identified colluders (malicious relays). Trust scores develop over time through the EWMA update (Eq. (4.3)), incorporating factors such as participation, correctness, delay, and penalties for challenges or collusion.

Interpretation: Honest relays generally demonstrate stable or progressively improving trust scores through consistent participation and accurate forwarding, whereas colluders display a decline in trust attributed to message drops, increased delays, and a higher challenge rate. The distinction between honest and malicious trajectories illustrates the EWMA's capacity to integrate both recent behavior and historical reputation, enabling the system to progressively demote misbehaving nodes.

Implications: The observed dynamics confirm that reputation serves as both informative and actionable: deteriorating behavior results in swift penalization (through the $(1-\alpha)\,\sigma(R)$ term), whereas consistent honesty either restores or maintains weight. This facilitates the utilization of T_i in Eq. (4.5) to regulate stake influence and the quorum decision-making process.

6.2.5 Figure 6.5: Reward histogram (final epoch)

$$R_i = f(w_i^{(t)}, \text{reward pool}, \text{epochal allocation}) \tag{6.1}$$

What the figure shows: This histogram displays the per-node reward distribution observed at the final epoch of the simulation. The distribution reflects the combined

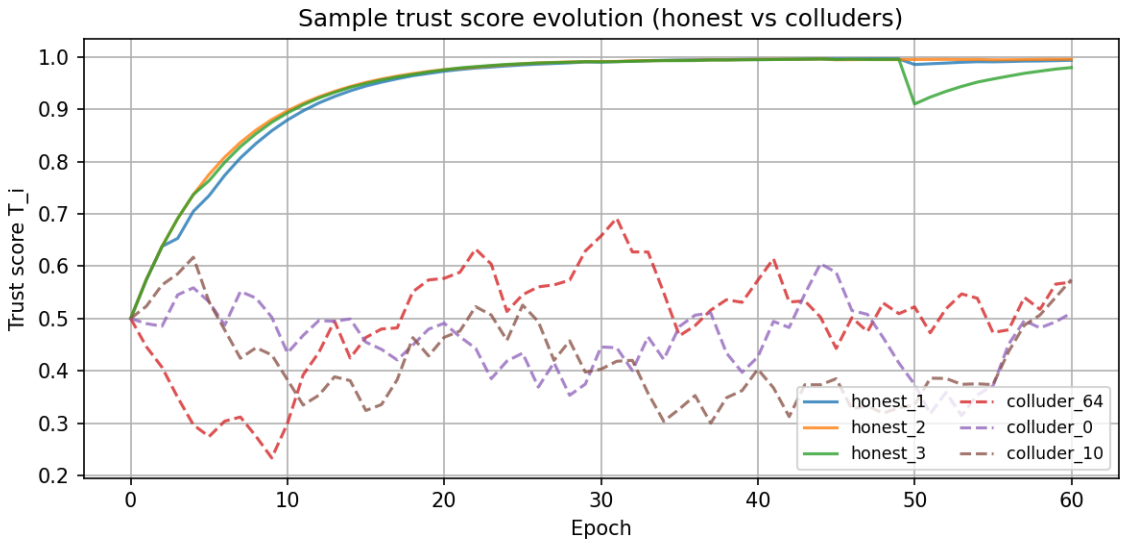

Figure 6.4: Time series of trust scores for all nodes over the course of the simulation. Honest nodes exhibit stable or gradually increasing trust, while malicious nodes experience a decline due to penalties, reflecting the system's ability to isolate and demote misbehaving participants.

effects of effective weight $w_i^{(t)}$, the reward pool, and the epochal allocation (Eq. (6.1)).

Interpretation: Rewards are concentrated among nodes that have accumulated both stake and reputation throughout the duration, indicated by high T_i and nontrivial $\widetilde{S}_i$ as per Eq. (4.4). The histogram shape, which is not significantly heavy-tailed, suggests that reputation-gating effectively prevents a limited number of high-stake yet low-reputation nodes from obtaining the majority of rewards.

Implications: The results support the claim that the hybrid stake and reputation weighting system generates incentive-compatible rewards: dependable relays with steady performance are rewarded, while malicious or inactive nodes receive minimal compensation. If reward centralization is a problem, measures such as minimum

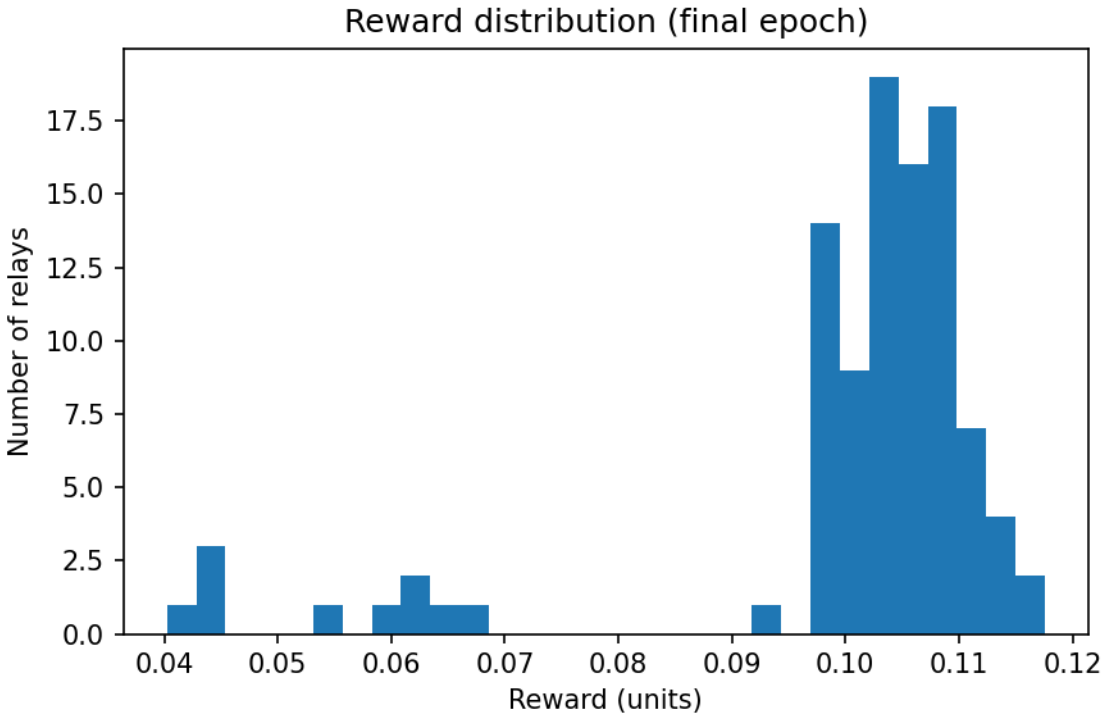

Figure 6.5: Histogram of cumulative rewards per node at the end of the simulation. The distribution shows that rewards are correlated with sustained positive trust scores, indicating that the reward mechanism effectively incentivizes honest participation.

payouts per epoch, reward smoothing, or light randomization in the initial epochs may be necessary.

6.2.6 Figure 6.6: Attacker success probability vs. adversary stake fraction

What the figure shows: The plot reports the empirical probability that an adversary controlling a given fraction of the total stake can assemble sufficient effective

weight to satisfy the quorum threshold τ and thereby finalize a malicious candidate.

Interpretation: The probability of success is minimal for stake fractions significantly below the quorum threshold and increases sharply as the adversary's stake nears the threshold. This behavior aligns with the quorum rule (Eq. (4.6)) and the principle of multiplicative reputation gating: attaining quorum necessitates both stake and reputation, thereby elevating the economic and temporal costs for potential attackers.

Implications: Under the simulated conditions, the architecture is resilient to single-shot stake attacks unless the attacker controls a large fraction of available stake. This supports the selection of τ within the suggested range (0.66–0.75) for a conservative deployment. In riskier threat models, increasing τ and/or requiring longer reputation accumulation windows can further harden the system.

6.2.7 Figure 6.7: Deanonymization probability vs. ring size

What the figure shows: The figure illustrates how likely an adversary is to correctly pinpoint the source of a transaction depending on the size of the ring signature denoted by r. For comparative clarity, the plot also incorporates a curve that approximates the impact of employing blind signatures, evidenced by a pronounced drop in the deanonymization probability, modeled here as a gain in unlinkability beyond the primary ring signature effect.

Interpretation: The deanonymization probability decreases as $1/r$ for ring signatures. Larger rings increase the anonymity set, making it less likely to guess correctly. Blind signatures decouple the signer's identity from the message, further lowering link-

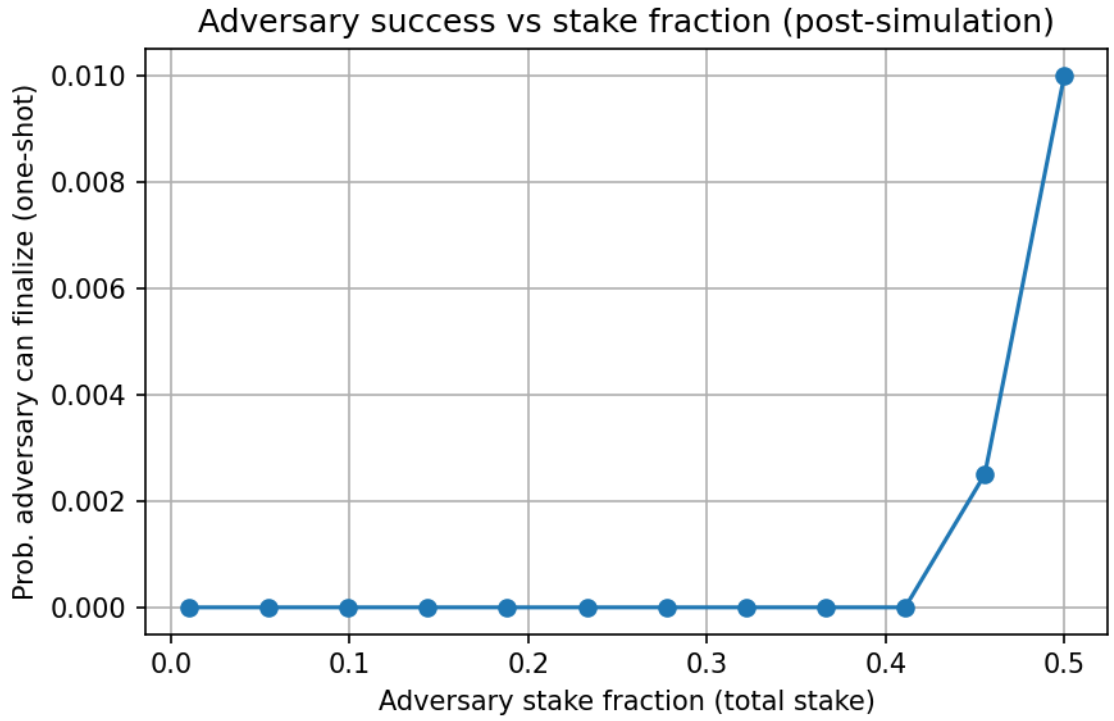

Figure 6.6: Attacker success probability as a function of total adversarial stake. The system demonstrates strong resistance to Sybil and majority stake attacks until adversaries control a critical stake threshold, beyond which compromise probability increases sharply.

age probability. This further lowers the chance of success for the adversary, especially with small to moderate ring sizes.

Implications: From a design viewpoint, increasing the ring size is a reliable way to improve cryptographic anonymity. Blind signatures help reduce signature-linkage attacks. For your specified latency budget (ring processing overhead), choose the ring size that provides an acceptable P_{correct} in your target threat model.

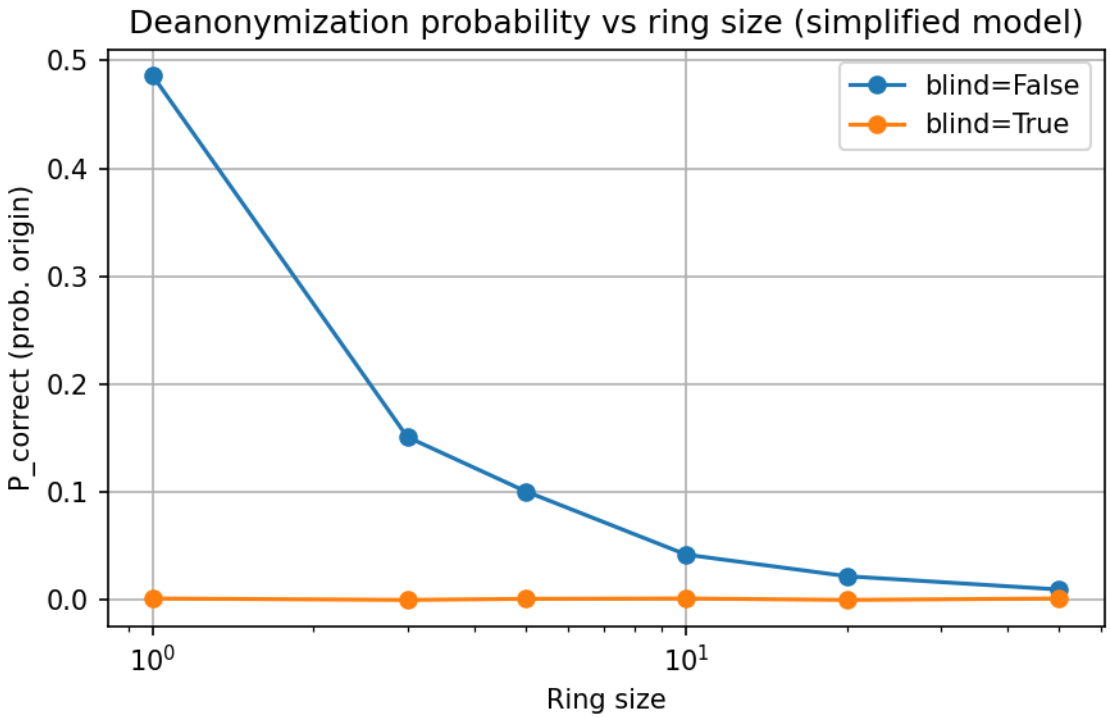

Figure 6.7: Deanonymization probability versus ring size in the ring signature scheme. Larger ring sizes significantly reduce the probability of correctly identifying the transaction origin, demonstrating the effectiveness of the cryptographic anonymity layer.

Chapter 7

Conclusion

This thesis presented a privacy-aware, incentive-driven bridge architecture to enable secure interoperability between a permissioned blockchain and a permissionless blockchain. The design combines (i) an anonymization-enabled relay cohort that forwards cross-chain messages using configurable source-anonymization protocols (Dandelion++ or Clover), (ii) cryptographic protections (ring signatures and blind signatures) to provide unlinkability and signer anonymity, and (iii) a reputation-aware incentive mechanism implemented as a Trust Contract that integrates per-epoch performance metrics with normalized stake to compute effective voting weights and epochal rewards. We evaluated the architecture through simulation experiments that stress-tested scalability and economic robustness, and systematically analyzed the privacy, correctness, and security properties.

7.1 Summary of contributions

The main contributions of this work are as follows:

- **Bridge architecture:** A blockchain-agnostic, relay-based architecture that mediates communication between permissioned and permissionless systems while minimizing additional trust assumptions and preserving permissioned-side confidentiality.

- **Privacy controls:** Integration of network-level anonymization (Dandelion++ and Clover) and cryptographic anonymity (ring and blind signatures) to reduce the probability of source deanonymization and to decouple relay identities from on-chain attestations.

- **Incentive and trust mechanism:** A compact, implementable trust model that aggregates observable epochal metrics (participation, delivery success, delay, challenge outcomes, and lightweight collusion indicators) using an EWMA update rule, combines reputation with concave stake normalization, and applies a weighted quorum rule for finalization and proportional reward allocation.

- **Security analysis and evaluation:** A threat-aware security analysis (Sybil, weighted-majority, collusion, front-running) and a set of simulation experiments that quantify the architecture's performance, anonymity guarantees, and resilience against economic and collusive attacks.

7.2 Key findings from the evaluation

The experimental campaign produced several findings that inform both the viability and the deployment trade-offs of the proposed system. The main outcomes are summarized below; each refers to results presented in Chapter 6.

7.2.1 Trust dynamics and reward fairness:

Time-series experiments of trust updates demonstrate that the EWMA-based trust update stabilizes honest relays' scores while penalizing intermittent misbehavior. The reward allocation, which is proportional to effective weights, yields concentrated but not overly skewed payouts: honest high-performing relays receive larger shares, while transient poor performance is rapidly reflected in reduced epochal rewards.

7.2.2 Resistance to stake- and Sybil-based attacks:

Parameter sweeps that varied adversarial stake and the number of Sybil identities indicate the multiplicative reputation gating and concave stake normalization slow down attempts to obtain controlling effective weight. In particular, the attacker-success vs. stake plots show that a rapidly acquired stake-only attack is ineffective unless accompanied by sustained reputation building—raising the time and economic cost for a successful takeover.

7.2.3 Privacy / deanonymization trade-offs:

Comparative experiments between Dandelion++ and Clover confirm the expected trade-off between anonymity and latency: longer stem/path lengths improve source obfuscation but increase end-to-end delay. Empirical deanonymization curves reveal that both protocols reduce deanonymization probability substantially compared to naive flooding. The precise advantage depends on path parameters and adversarial fraction. Ring signatures and blind-sign issuance further reduce linkability at the application layer.

7.2.4 Performance and scalability:

Latency cumulative distribution function (CDF) and parametric heatmaps demonstrate that the system scales reasonably with hundreds of relays under realistic simulated network conditions. The major contributors to end-to-end latency are anonymization path length and relay-induced delays; the Bridge and Trust Contracts themselves contribute minimal on-chain processing overhead in our simulated setting.

7.2.5 Equity and concentration metrics:

Gini-index analyses and reward histograms indicate that, while rewards are naturally concentrated toward more reliable relays, careful parameter choices (stake normalization, reputation floor ω_R, EWMA memory α) keep concentration within acceptable bounds to discourage centralization.

7.3 Limitations

While promising, this study has a few limitations that temper the conclusions and highlight directions for future work:

- **Simulation vs. deployment:** All evaluations were conducted in a simulated environment. Real-world networks introduce complex, correlated failures, network heterogeneity, and adversaries with richer strategies (bribery, targeted infiltration) that may materially change outcomes.

- **Simplified adversary models:** The experiments model adversaries by controlling a fraction of relays or stake. More sophisticated attacks—long-range stake-grinding, bribery, dynamic stake migration—require further economic and game-theoretic analysis.

- **Parameter sensitivity:** Security and performance depend on choices of protocol parameters (e.g., ω_R, τ, EWMA α, anonymization path length). While we recommend ranges supported by experiments, optimal tuning requires context-specific calibration.

- **On-chain costs and practical integration:** The Trust Contract assumes modest on-chain operations. In production environments, gas costs and smart-contract limitations (e.g., floating-point arithmetic, storage) will influence concrete implementations and may necessitate additional off-chain aggregation primitives.

7.4 Practical implications and recommendations

For practitioners considering adoption of the architecture, the evaluation suggests the following practical guidance:

- Use trust-weighted selection and reward schemes to prioritize reliability while preventing stake-only takeovers; enforce a reputation floor (ω_R) that protects against rapid stake-based influence.
- Tune anonymization parameters according to application SLAs: use shorter paths or selective anonymization for latency-sensitive messages, and fuller anonymization for privacy-critical transfers.
- Employ off-chain auditors or multiparty auditors to compute heavy statistics (collusion indicators, pairwise agreement) and commit compact, signed summaries on-chain to keep gas costs low while maintaining verifiability.
- Adopt epoch-snapshotting for quorum collection to mitigate timing-based front-running and replay attacks.

7.5 Final remarks

This thesis demonstrates that privacy-preserving cross-chain interoperability between permissioned and permissionless systems is feasible when cryptographic anonymity, network-level source obfuscation, and economically-aligned incentives are combined in a coherent architecture. The proposed trust model and weighted-quorum mechanism provide a pragmatic balance between safety and liveness, while anonymization

layers materially reduce deanonymization risk.

Chapter 8

Future Works

This thesis introduces a relay-based interoperability framework that connects a permissioned blockchain with a permissionless blockchain, incorporating enhanced privacy guarantees and incentive structures to promote the honest participation of bridge members. The design and evaluation also highlight promising directions for future research, offering opportunities for further expansion, refinement, and practical implementation. In particular, we outline four directions: (i) scalability and performance optimization, (ii) security hardening and adversarial models, (iii) privacy and anonymization enhancements, and (iv) governance and trust frameworks, all of which can guide future efforts to strengthen the robustness, efficiency, and real-world applicability of relay-based blockchain interoperability.

8.1 Scalability and Performance Optimization

- **Large-scale deployments of relays:**

Current evaluations rely on relatively small relay sets. Future work should examine how the system behaves with thousands of relays ($n \geq 1000$), measuring throughput, latency, and fault tolerance under realistic churn and adversarial conditions. Such studies would highlight scalability limits and guide optimizations for production-grade environments.

- **Hierarchical architectures:**

  Using multi-tier or sharded relay cohorts could reduce quorum collection costs and enhance fault isolation. A hierarchical design might allow localized subquorums to finalize messages within regions before forwarding them to higher-level layers, improving efficiency while maintaining global security.

- **Adaptive quorum thresholds:**

  The system could change the thresholds dynamically instead of fixing τ in response to real-time factors such as network congestion, observed adversarial participation, or latency constraints. This adaptivity would balance safety and liveness, ensuring the bridge remains functional under diverse conditions.

- **Resource-efficient computation:**

  Techniques such as succinct zero-knowledge proofs (SNARKs/STARKs) could make off-chain trust computations more compact and verifiable on-chain at a fraction of the cost. This would be particularly beneficial in high-frequency bridge activity scenarios, where repeated trust updates might otherwise incur prohibitive gas costs.

8.2 Security Hardening and Adversarial Models

- **Expanded adversarial strategies:**

  The current analysis examines Sybil, weighted-majority, and collusion attacks. Subsequent research with a broader adversarial model should include eclipse attacks, long-range stake-grinding, bribery attacks, and front-running strategies in opposition to relay incentives. Testing the bridge against these attacks would provide stronger confidence in its robustness under real-world adversarial conditions.

- **Game-theoretic analysis:**

  A deeper investigation into incentive compatibility using game-theoretic tools could ensure that honest behavior remains not just preferable but strictly dominant under varied strategic settings. Formal equilibrium proofs and stress testing across heterogeneous relay populations would further strengthen the mechanism's reliability.

- **Post-quantum resilience:**

  Since the bridge heavily depends on digital signatures and hashing, adopting quantum-resistant primitives such as lattice-based schemes will be essential for long-term security. Evaluating migration pathways to post-quantum cryptography ensures the bridge remains viable in the face of future cryptanalytic advances.

8.3 Privacy and Anonymization Enhancements

- **Comparative anonymization protocols:**

  While Dandelion++ and Clover were considered, integrating alternative protocols such as mixnets Chaum [1981] or onion routing could provide stronger anonymity under global adversaries capable of large-scale traffic analysis. A comparative study of these approaches would clarify trade-offs between latency, overhead, and privacy.

- **Hybrid privacy models:**

  Investigating selective anonymization that only hides sensitive cross-chain. This approach would reduce overhead while ensuring strong privacy protections where most needed, achieving a more balanced privacy–performance trade-off.

- **Formal anonymity metrics:**

  Developing rigorous deanonymization metrics and empirically evaluating them under different adversarial models (e.g., partial network observation, timing analysis) would make privacy guarantees more precise, reproducible, and comparable across different protocols.

8.4 Governance and Trust Frameworks

- **Auditor roles:**

  he off-chain auditor currently acts as a centralized auxiliary trust provider. Future work could investigate decentralized governance models, such as auditor

committees elected by token holders, rotating assignments among relays, or marketplaces where independent auditors compete to provide verifiable oversight. These models would reduce reliance on any single entity.

- **Decentralized slashing appeals:**

  Transparent dispute-resolution processes would help avoid unjust penalties. Mechanisms inspired by optimistic rollup fraud proofs or decentralized arbitration systems could allow relays to contest slashing events, thereby increasing fairness while preserving accountability.

Bibliography

M. Alharby and A. Van Moorsel. Blockchain-based smart contracts: A systematic mapping study. *arXiv preprint arXiv:1710.06372*, 2017.

O. Aluko and A. Kolonin. Proof-of-reputation: an alternative consensus mechanism for blockchain systems. *arXiv preprint arXiv:2108.03542*, 2021.

E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. D. Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolić, S. W. Cocco, and J. Yellick. Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. In *Proceedings of the Thirteenth EuroSys Conference*, pages 1–15, Apr. 2018. doi: 10.1145/3190508.3190538. URL `http://arxiv.org/abs/1801.10228`. arXiv:1801.10228 [cs].

S. Bano, A. Sonnino, M. Al-Bassam, S. Azouvi, P. McCorry, S. Meiklejohn, and G. Danezis. Sok: Consensus in the age of blockchains. In *Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, pages 183–198, 2019.

F. Barbàra and C. Schifanella. MP-HTLC: Enabling blockchain interoperability through a multiparty implementation of the hash time-lock contract. *Concurrency*

and Computation: Practice and Experience, 35(9):e7656, Apr. 2023. ISSN 1532-0626, 1532-0634. doi: 10.1002/cpe.7656. URL https://onlinelibrary.wiley.com/doi/10.1002/cpe.7656.

R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia. A Survey on Blockchain Interoperability: Past, Present, and Future Trends. *ACM Computing Surveys*, 54 (8):1–41, Nov. 2022. ISSN 0360-0300, 1557-7341. doi: 10.1145/3471140. URL https://dl.acm.org/doi/10.1145/3471140.

S. Bojja Venkatakrishnan, G. Fanti, and P. Viswanath. Dandelion: Redesigning the bitcoin network for anonymity. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 1(1):1–34, 2017.

L. Breidenbach, C. Cachin, A. Coventry, S. Ellis, A. Juels, A. Miller, B. Magauran, S. Nazarov, A. Topliceanu, F. Zhang, B. Chan, F. Koushanfar, D. Moroz, and F. Tramer. Chainlink 2.0: Next Steps in the Evolution of Decentralized Oracle Networks. *Preprint*, 2021.

R. G. Brown, J. Carlyle, I. Grigg, and M. Hearn. Corda: an introduction. *R3 CEV, August*, 1(15):14, 2016.

D. Chaum. Blind signatures for untraceable payments. In *Advances in Cryptology: Proceedings of Crypto 82*, pages 199–203. Springer, 1983.

D. L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.

H. Chen, M. Pendleton, L. Njilla, and S. Xu. A survey on ethereum systems security:

Vulnerabilities, attacks, and defenses. *ACM Computing Surveys (CSUR)*, 53(3): 1–43, 2020.

G. Ciccarelli and R. L. Cigno. Collusion in peer-to-peer systems. *Computer Networks*, 55(15):3517–3532, 2011.

Consensys. Quorum: An enterprise-focused version of Ethereum. `https://consensys.net/quorum/`, 2023. Accessed: 2025-07-07.

M. de Vos, C. U. Ileri, and J. Pouwelse. Xchange: A universal mechanism for asset exchange between permissioned blockchains. *World Wide Web*, 24(5):1691–1728, 2021.

Z. Deng, C. Tang, T. Li, P. Abla, Q. Chen, W. Liang, and D. He. Enhancing blockchain cross chain interoperability: A comprehensive survey. *arXiv preprint arXiv:2505.04934*, 2025.

T. T. A. Dinh, A. Datta, and B. C. Ooi. A blueprint for interoperable blockchains. *arXiv preprint arXiv:1910.00985*, 2019.

J. R. Douceur. The sybil attack. In *International workshop on peer-to-peer systems*, pages 251–260. Springer, 2002.

European Data Protection Supervisor. AEPD–EDPS Joint Paper on Anonymisation, April 2021. URL `https://www.edps.europa.eu/system/files/2021-04/21-04-27_aepd-edps_anonymisation_en_5.pdf`. Accessed: 2025-05-21.

G. Fanti and P. Viswanath. Deanonymization in the bitcoin p2p network. *Advances in neural information processing systems*, 30, 2017.

G. Fanti, S. B. Venkatakrishnan, S. Bakshi, B. Denby, S. Bhargava, A. Miller, and P. Viswanath. Dandelion++: Lightweight Cryptocurrency Networking with Formal Anonymity Guarantees. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 2(2):1–35, June 2018. ISSN 2476-1249. doi: 10.1145/3224424. URL `https://dl.acm.org/doi/10.1145/3224424`.

H. Foundation. Hyperledger cacti: A pluggable blockchain interoperability framework. `https://hyperledger.org/projects/cacti`, 2025. Accessed: 2025-01-04.

F. Franzoni and V. Daza. Clover: an Anonymous Transaction Relay Protocol for the Bitcoin P2P Network, Sept. 2021. URL `http://arxiv.org/abs/2109.00376`. arXiv:2109.00376 [cs].

P. Frauenthaler, M. Sigwart, C. Spanring, M. Sober, and S. Schulte. ETH Relay: A Cost-efficient Relay for Ethereum-based Blockchains. In *2020 IEEE International Conference on Blockchain (Blockchain)*, pages 204–213, Rhodes Island, Greece, Nov. 2020. IEEE. ISBN 978-0-7381-0495-9. doi: 10.1109/Blockchain50366.2020.00032. URL `https://ieeexplore.ieee.org/document/9284781/`.

GDPR2016. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation). `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32016R0679`, April 2016. Official Journal of the European Union, L119/1, 4 May 2016. Accessed: 2025-05-21.

P. Han, Z. Yan, L. T. Yang, and E. Bertino. P2c2t: Preserving the privacy of cross-

chain transfer. In *2025 IEEE Symposium on Security and Privacy (SP)*, pages 1474–1492. IEEE, 2025.

T. Hardjono, A. Lipton, and A. Pentland. Toward an Interoperability Architecture for Blockchain Autonomous Systems. *IEEE Transactions on Engineering Management*, 67(4):1298–1309, Nov. 2020. ISSN 0018-9391, 1558-0040. doi: 10.1109/TEM.2019.2920154. URL https://ieeexplore.ieee.org/document/8743548/.

J. Huang, K. Lei, M. Du, H. Zhao, H. Liu, J. Liu, and Z. Qi. A survey on blockchain incentive mechanism. In *Data Science and Big Data*, volume 1085 of *Communications in Computer and Information Science*, pages 386–395. Springer, 2019. doi: 10.1007/978-981-15-0118-0_30. URL https://doi.org/10.1007/978-981-15-0118-0_30.

J. S. Hunter. The exponentially weighted moving average. *Journal of quality technology*, 18(4):203–210, 1986.

C. Ihle, D. Trautwein, M. Schubotz, N. Meuschke, and B. Gipp. Incentive mechanisms in peer-to-peer networks—a systematic literature review. *ACM Computing Surveys*, 55(14s):1–69, 2023.

A. Kamilaris, A. Fonts, and F. X. Prenafeta-Boldύ. The rise of blockchain technology in agriculture and food supply chains. *Trends in Food Science & Technology*, 91:640–652, Sept. 2019. ISSN 09242244. doi: 10.1016/j.tifs.2019.07.034. URL https://linkinghub.elsevier.com/retrieve/pii/S0924224418303686.

G. Kappos, H. Yousaf, M. Maller, and S. Meiklejohn. An empirical analysis of

anonymity in zcash. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 463–477. USENIX Association, 2018.

A. Kiayias and P. Lazos. Sok: blockchain governance. In *Proceedings of the 4th ACM Conference on Advances in Financial Technologies*, pages 61–73, 2022.

J. Kwon and E. Buchman. Cosmos: A network of distributed ledgers. `https://cosmos.network/whitepaper`, 2019. Accessed: 2025-01-04.

T. Li, Y. Chen, Y. Wang, Y. Wang, M. Zhao, H. Zhu, Y. Tian, X. Yu, and Y. Yang. Rational protocols and attacks in blockchain system. *Security and Communication Networks*, 2020(1):8839047, 2020.

X. Li, Q. Liu, S. Wu, Z. Cao, and Q. Bai. Game theory based compatible incentive mechanism design for non-cryptocurrency blockchain systems. *Journal of Industrial Information Integration*, 31:100426, 2023.

Y. Liu, Q. Lu, L. Zhu, H.-Y. Paik, and M. Staples. A systematic literature review on blockchain governance. *Journal of Systems and Software*, 197:111576, 2023.

Z. Liu, N. C. Luong, W. Wang, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim. A survey on applications of game theory in blockchain. *arXiv preprint arXiv:1902.10865*, 2019.

S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G. M. Voelker, and S. Savage. A fistful of bitcoins: Characterizing payments among men with no names. In *Proceedings of the 2013 Internet Measurement Conference (IMC)*, pages

127–140. ACM, 2013. doi: 10.1145/2504730.2504747. URL https://cseweb.ucsd.edu/~smeiklejohn/files/imc13.pdf.

B. K. Mohanta, S. S. Panda, and D. Jena. An overview of smart contract and use cases in blockchain technology. In *2018 9th international conference on computing, communication and networking technologies (ICCCNT)*, pages 1–4. IEEE, 2018.

Monika and R. Bhatia. Interoperability Solutions for Blockchain. In *2020 International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, pages 381–385, Bengaluru, India, Oct. 2020. IEEE. ISBN 978-1-72817-213-2. doi: 10.1109/ICSTCEE49637.2020.9277054. URL https://ieeexplore.ieee.org/document/9277054/.

M. Moser, R. Böhme, and D. Breuker. An empirical analysis of traceability in the monero blockchain. *Proceedings on Privacy Enhancing Technologies*, 2018(3):143–163, 2018.

S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf, n.d. Accessed: 2025-01-04.

A. Network. Axelar: Decentralized interoperability protocol. *White Paper*, 2021. URL https://axelar.network/whitepaper.pdf.

D. Ongaro and J. Ousterhout. In search of an understandable consensus algorithm. In *2014 USENIX annual technical conference (USENIX ATC 14)*, pages 305–319, 2014.

R. v. Pelt, S. Jansen, D. Baars, and S. Overbeek. Defining blockchain governance: A

framework for analysis and comparison. *Information Systems Management*, 38(1): 21–41, 2021.

C. Pinzón and C. Rocha. Double-spend attack models with time advantange for bitcoin. *Electronic Notes in Theoretical Computer Science*, 329:79–103, 2016.

M. Raskin. The law and legality of smart contracts. *Geo. L. Tech. Rev.*, 1:305, 2016.

K. Ren, N.-M. Ho, D. Loghin, T.-T. Nguyen, B. C. Ooi, Q.-T. Ta, and F. Zhu. Interoperability in Blockchain: A Survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12750–12769, Dec. 2023. ISSN 1041-4347, 1558-2191, 2326-3865. doi: 10.1109/TKDE.2023.3275220. URL `https://ieeexplore.ieee.org/document/10123097/`.

R. L. Rivest, A. Shamir, and Y. Tauman. How to leak a secret. In *Proceedings of ASIACRYPT*, 2001.

A. Shostack. *Threat Modeling: Designing for Security.* Wiley, Hoboken, NJ, 2014. ISBN 978-1-118-80999-0.

Y. Sun, L. Yi, L. Duan, and W. Wang. A Decentralized Cross-Chain Service Protocol based on Notary Schemes and Hash-Locking. In *2022 IEEE International Conference on Services Computing (SCC)*, pages 152–157, Barcelona, Spain, July 2022. IEEE. ISBN 978-1-66548-146-5. doi: 10.1109/SCC55611.2022.00033. URL `https://ieeexplore.ieee.org/document/9860209/`.

N. Szabo. Smart contracts. `https://www.fon.hum.uva.nl/rob/Courses/`

InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart.contracts.html, 1994.

C. Times. The stealth revolution: How dandelion redefines anonymity in crypto. https://www.cypherpunktimes.com/the-stealth-revolution-how-dandelion-redefines-anonymity-in-crypto/, 2023. Accessed: 2025-08-18.

P. Tolmach, Y. Li, S.-W. Lin, Y. Liu, and Z. Li. A survey of smart contract formal specification and verification. *ACM Computing Surveys (CSUR)*, 54(7):1–38, 2021.

M. Tuler de Oliveira and Others. Blockchain reputation-based consensus: A scalable and resilient mechanism for distributed mistrusting applications. *Computer Networks*, 179, 2020. doi: 10.1016/j.comnet.2020.107367.

G. Wood. Polkadot: Vision for a heterogeneous multi-chain framework. In *White paper*, volume 21, page 4662, 2016.

X. Xu, I. Weber, and M. Staples. Architecture for blockchain applications. *Journal Name Placeholder*, 2019a.

X. Xu, I. Weber, M. Staples, L. Zhu, J. Bosch, L. Bass, C. Pautasso, and P. Rimba. A taxonomy of blockchain-based systems for architecture design. *IEEE Software*, 36(1):70–77, 2019b.

H. Yuan. Two major attacks on the ethereum smart contract application layer. In *Third International Conference on Machine Learning and Computer Application (ICMLCA 2022)*, volume 12636, pages 1230–1237. SPIE, 2023.

A. Zamyatin, D. Harz, J. Lind, P. Panayiotou, A. Gervais, and W. Knottenbelt. Xclaim: Trustless, interoperable, cryptocurrency-backed assets. In *2019 IEEE symposium on security and privacy (SP)*, pages 193–210. IEEE, 2019.

M. Zhang, X. Zhang, Y. Zhang, and Z. Lin. Txspector: Uncovering attacks in ethereum from transactions. In *Proceedings of the 29th USENIX Security Symposium (USENIX Security 20)*, pages 2775–2792, 2020.