Ika: Fast, Decentralized and Secure Interoperability Powered by 2PC-MPC

Omer Sadika, David Lachmish, Yehonatan Cohen Scaly, and Sheran Hussain

Ika Team

Version 1.0, October 2024

Abstract

We propose Ika, a distributed system of MPC signers that can be used as a comprehensive solution to blockchain interoperability challenges through a Zero Trust cryptographic framework paired with an architecture optimized for performance and decentralization. Ika integrates the novel 2PC-MPC protocol, a cutting-edge advancement in threshold cryptography, alongside state-of-the-art DAG-based Byzantine consensus protocol Mysticeti as the broadcast channel to achieve reliable broadcast. The system can enable secure, low-latency, and decentralized cross-chain interactions, and establish a new standard for blockchain interoperability.

1 Introduction

Blockchain technology has revolutionized digital assets and the way decentralized applications are built, enabling trustless interactions and disintermediated financial systems. However, the rapid expansion of blockchain ecosystems has exposed critical challenges in interoperability, scalability, and decentralization.

At the heart of these challenges is the lack of secure and efficient cross-chain communication. Current solutions such as cross-chain bridges, centralized custodians, and federated Multi-Party Computation (MPC) systems often trade security or performance for functionality. These architectures fail to meet the rigorous requirements of a truly decentralized system: eliminating single points of failure, achieving low-latency, high-throughput operations, and ensuring scalability without compromising security.

Current blockchain interoperability systems face significant challenges, including security risks such as collusion and honeypots; performance limitations caused by high latency and low throughput that hinder real-time, high-volume use cases; and decentralization constraints that compromise resilience and censorship resistance.

To address these issues, blockchain interoperability requires a solution that is secure, highly performant, and scalable while adhering to decentralization principles. This involves a **Zero Trust architecture** to eliminate reliance on trusted intermediaries, an infrastructure capable of **low-latency and high-throughput** operations to support modern use cases, and a system design that scales with **decentralized participation**, enabling robust, trustless collaboration at scale.

This whitepaper provides a structured high-level analysis of the problem landscape, existing solutions, and our proposed technical architecture for Ika. It also explores real-world use cases and governance mechanisms, demonstrating how Ika can create a secure, scalable, and decentralized foundation for the next generation of blockchain applications.

2 The Problem Landscape

Blockchain ecosystems have grown into complex and diverse networks, yet they remain fundamentally isolated. The lack of secure, efficient, and decentralized interoperability mechanisms has created a fragmented environment where assets and data cannot move freely without significant risk. This challenge has given rise to solutions like cross-chain bridges, federated Multi-Party Computation (MPC) systems, and centralized custodians. However, these approaches often fall short, introducing critical tradeoffs in security, performance, and decentralization[1].

At the heart of the interoperability problem is the issue of trust. Cross-chain systems typically rely on intermediaries, whether in the form of validators, federated groups, or centralized entities. These intermediaries create single points of failure, exposing the system to risks such as collusion, censorship, and

outright theft. For example, in federated MPC systems, a small set of signers collectively manage assets and approve transactions. While this setup can reduce operational complexity, it comes at the cost of security; they require users to award control of their assets to that small set of signers, a stark contradiction to the ethos of decentralization and blockchain technology, and if enough participants collude, the assets can be drained.

Another critical vulnerability arises from the inherent complexity of synchronization between blockchains. Cross-chain bridges, for instance, rely on locking assets on one blockchain and issuing synthetic representations - wrapped tokens[2] - on another. This process not only increases the attack surface but also introduces significant operational risks, such as replay attacks or double-spending. These vulnerabilities have been exploited in numerous high-profile incidents, where billions of dollars in digital assets have been lost[3].

Performance is another area where traditional interoperability solutions falter. Many existing systems struggle to deliver low-latency, high-throughput operations essential for real-time applications. Protocols that were designed for centralized custodians and are used by federated MPC systems, often introduce delays that make them impractical for use cases requiring rapid transaction settlement. Limited scalability further exacerbates the issue; as demand grows, these systems face bottlenecks, leading to degraded performance and higher costs.

Equally concerning is the erosion of decentralization in many interoperability frameworks[4]. Federated systems typically operate with a small number of parties - often fewer than 20 - leading to centralization of control. This not only undermines the security of the system but also exposes it to censorship risks, where dominant participants can exclude transactions or manipulate outcomes. Such architectures fail to align with the core principles of blockchain, which emphasize trustless, permissionless, and resilient systems.

Moreover, existing solutions fail to achieve true native interoperability. Instead of enabling direct interactions between blockchains, they rely on wrapped tokens or external mechanisms to facilitate cross-chain activity. This approach not only complicates transaction workflows but also introduces additional layers of risk. For example, the reliance on bridges requires extensive coordination between blockchains, increasing the likelihood of synchronization errors or smart contract vulnerabilities.

These issues are not isolated challenges but interconnected limitations that have collectively stifled the potential of blockchain ecosystems. Security, performance, and decentralization are often treated as trade-offs rather than fundamental requirements, forcing developers and users to make compromises that hinder adoption and innovation.

The current landscape makes it clear that a new approach is needed - one that eliminates the need for trust in intermediaries, scales efficiently with demand, and upholds the principles of decentralization. The next section will introduce Ika's Zero Trust cryptographic framework, which addresses these challenges comprehensively, delivering a secure, high-performance, and decentralized solution for blockchain interoperability.

3 IKA's Solution

3.1 dWallet: a New Primitive

A *dWallet* is a transformative building block for the decentralized ecosystem. A dWallet serves as a programmable infrastructure component that developers can integrate into their smart contracts, enabling highly flexible and decentralized applications. A dWallet controls accounts on different chains (e.g. Sui, Bitcoin, Ethereum, Arbitrum, Solana etc.) by generating valid signatures for the corresponding account addresses, allowing smart contracts to enforce specific conditions under which a signature can be generated. By doing so the dWallet primitive enables smart contracts to control addresses and sign arbitrary transactions to other chains.

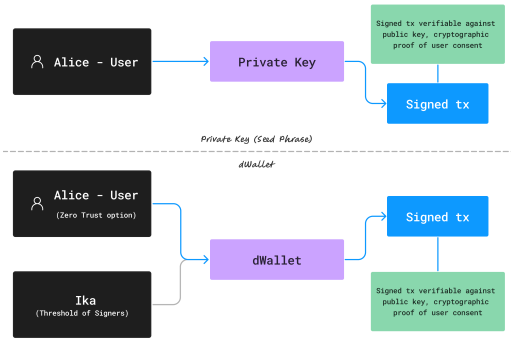

Figure 1: Cryptographic Verification of User Participation

Just like a signed transaction in a blockchain's state is a cryptographic proof of user participation when using a private key, so is a dWallet-signed transaction.

At its core, a dWallet divides its "private signing key" into cryptographic shares managed by the

network with the option of cryptographically requiring user participation (Zero Trust[5]). This ensures that no single participant can independently use or control the key. Any operation, such as signing a transaction, requires secure collaboration within the network signers (and in a Zero Trust option also the user). Ika employs economic security to incentivize honest actors similarly to blockchains, and the Zero Trust option safeguards the dWallet further, even in scenarios where some network participants are compromised.

A critical feature of the dWallet is its programmability, dWallets are designed to execute complex logic in collaboration with smart contracts. Developers can use dWallets to implement application-specific behaviors, such as enforcing spending limits, creating time-locked transactions, or requiring multi-party approval for high-value transactions. For example, a smart contract for decentralized finance (DeFi) could interact with a dWallet to execute predefined trading strategies or manage liquidity pools between different chains.

In addition to programmability, dWallets support dynamic participation, allowing network nodes to join or leave without disrupting the wallet's functionality. This flexibility ensures resilience and scalability, making dWallets suitable for a wide range of use cases, from personal wallets to enterprise-grade applications. Developers can rely on this dynamic architecture to build systems that adapt to changing conditions, ensuring reliability in large-scale, distributed environments.

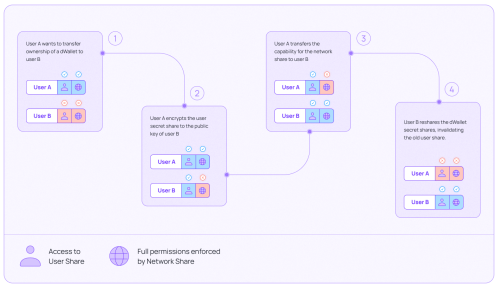

Figure 2: dWallet Transfer

A simple implementation for a protocol handling dWallet transfers - revealing the user share to the new owner, granting them the capability to control the network share, and initiating a reshare of the dWallet secret shares by the new owner.

One of the most powerful features of the dWallet is its transferability. A dWallet can be securely transferred between users or integrated into smart contracts. This capability unlocks significant opportunities for developers. For instance, a DAO (Decentralized Autonomous Organization) could transfer control of a treasury-managed dWallet to a newly elected governing body, or a DeFi protocol could programmatically reassign ownership of a collateralized wallet based on predefined conditions. This ability to transfer accounts, introduced for the first time by Ika, opens up versatile new use cases for blockchain applications.

The operation of a dWallet consists of two primary functionalities. First is Distributed Key Generation (DKG), where the secret shares of the dWallet are cryptographically and securely generated by the user and the network, and the public key from which the account addresses on different chains are derived is also calculated. Ika allows replacing the DKG with the import of an existing private key - in this scenario the user holds the full original private key, so it cannot be used in use cases where external logic must be enforced (e.g. liquidation). Second is transaction signing, where the user collaborates with the network to generate a signature without ever reconstructing the full private key. dWallets can be bound by externally defined programmatic logic, which can be invoked by smart contracts. This architecture ensures that a dWallet functions as a trustless, programmable and transferable asset management tool, seamlessly integrating with decentralized applications.

dWallets represent a paradigm shift in blockchain infrastructure. They are not merely tools for securing digital assets; they are programmable building blocks that redefine how assets and logic interact in decentralized systems. By combining cryptographic decentralization, programmability, dynamic participation, and transferability, dWallets empower developers to build sophisticated, secure, and resilient applications.

With this foundational understanding of dWallets, we now turn to the cryptographic engine that powers them. The next section explores the 2PC-MPC protocol, the cornerstone of Ika's architecture, and its role in enabling the secure and collaborative functionality of dWallets.

3.2 The 2PC-MPC Protocol

The 2PC-MPC protocol[6] is the backbone of Ika's cryptographic framework. By enabling collaborative signing processes between a user and a decentralized network, the protocol eliminates trust in individual participants while ensuring robust security and performance. The 2PC-MPC protocol, integrated with Mysticeti[7], leverages broadcast communication in-

stead of unicast, enabling high scalability and performance even as the number of participants grows.

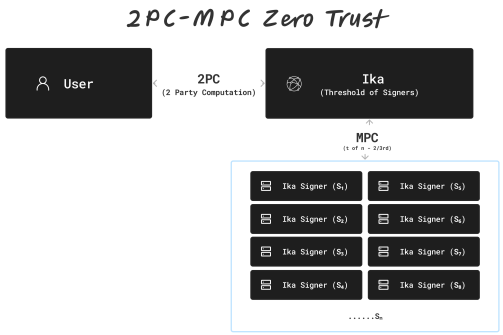

Figure 3: 2PC-MPC Zero Trust Architecture
Every signature requires both the user and a decentralized network to participate

3.2.1 Distributed Key Generation and Threshold Security

In the 2PC-MPC framework, two secret shares are generated by the user and the network in a DKG process (2PC: 2 Party Computation). One share is retained by the user, while the second share is encrypted on the network using homomorphic encryption, allowing a threshold of the network nodes to operate it (MPC: Multi Party Computation).

This structure prevents any single party, including the network participants, from reconstructing the private key. This ensures that security is distributed across the network, protecting against insider threats or external compromise. In cases where the user participation is not preferable, the user share can be stored publicly, such that anyone can generate the signature together with the network, making it redundant, effectively giving full control of the dWallet to the smart contract enforcing the logic.

3.2.2 Presignatures and Transaction Signing

The protocol uses a presignature mechanism to enhance performance. Presignatures[8] are partially computed signatures, independent of the message to be signed, that can be precomputed by the network nodes. These precomputed elements allow the signing process to be highly efficient when a user initiates a transaction.

When a transaction request is made, the user submits their partial signatures (cryptographic data) and the message to be signed. The network calculates the signature homomorphically over the encrypted network share, and combines it with the user input and a precomputed presignature which results in an encrypted valid signature, which will be decrypted by a threshold of network participants (threshold homomorphic decryption - MPC). This description is a simplification of a complex and robust cryptographic process (read the full 2PC-MPC paper[9] for a detailed breakdown) that is designed to resist many attack vectors, to assure that at no point is the private key reconstructed, ensuring robust security throughout the operation.

3.2.3 Broadcast Communication and Scalability

A key innovation in Ika's implementation of 2PC-MPC is its reliance on broadcast communication. Instead of unicast communication, which leads to message complexity of $O(n^2)$, as each participant communicates directly with every other participant, the protocol uses the blockchain's reliable broadcast communication channels to communicate across the network. This design not only fits the blockchain design more naturally but also dramatically improves scalability, as messages are sent to the entire network, reducing message complexity to $O(n)$. To further enhance performance, batching and amortization techniques are employed. Batching proving and verification together and amortization spreads the computational cost across multiple operations.

Removing proofs from the blockchain side, and emulating the two-party communication in MPC reduces complexity from the point of view of the user to practically $O(1)$, enabling efficient handling of real-time, large-scale computations. This makes Ika particularly well-suited for decentralized finance (DeFi), custody solutions, and cross-chain interoperability.

The reliable broadcast channel in Ika is implemented using the Mysticeti consensus protocol that is used on the Sui blockchain (following the recent upgrade from Narwhal[10] and Bullshark[11]). Mysticeti employs a Directed Acyclic Graph (DAG)-based approach[12], which ensures low latency and high throughput even under adverse network conditions. The DAG structure allows multiple messages to be processed simultaneously, ensuring that the broadcast system can scale efficiently as the network grows. With Mysticeti, Ika achieves an optimal balance between latency and performance, committing operations with minimal communication overhead.

3.2.4 Asynchronous and Fault-Tolerant Design

The 2PC-MPC protocol operates in asynchronous environments, where network participants may join or leave dynamically. This is achieved through a quorum-based design, ensuring that transactions can proceed as long as a sufficient subset of nodes is available. The use of Mysticeti for broadcast communication further enhances fault tolerance, allowing the protocol to maintain high performance even when faced with partial network failures or Byzantine faults.

Mysticeti's DAG-based consensus mechanism ensures that the protocol can process transactions efficiently without requiring global synchronization. By minimizing the number of communication rounds and leveraging a threshold logical clock, Mysticeti allows the 2PC-MPC protocol to achieve optimal latency and throughput.

3.2.5 Security and Verifiability

The security guarantees of the 2PC-MPC protocol are grounded in its cryptographic design. Public verifiability ensures that all operations are auditable, while identifiable abort mechanisms allow the network to detect and address malicious behavior. The integration with Mysticeti further reinforces these guarantees by providing a reliable and tamper-resistant broadcast channel.

By design, the protocol ensures that secret shares remain secure and distributed, protecting against both insider and external threats. Additionally, the use of threshold homomorphic encryption ensures that sensitive computations are performed without exposing their underlying data.

2PC-MPC also has guaranteed output, which means that even if there are malicious parties or a party goes offline mid-session, as long as the required threshold participates honestly, there is a cryptographic assurance that the session will be concluded successfully.

3.2.6 Real-World Application: A Transaction Lifecycle

Consider a scenario where a user initiates a transaction using their dWallet. The user generated partial signature and the transaction data are submitted to the network. The network nodes, using the user input and precomputed presignatures communicate over the Mysticeti reliable broadcast channel to generate a valid signature. The resulting signature is then used to submit the transaction to the target blockchain (e.g. Bitcoin), with the entire process completed without ever reconstructing the private key.

This architecture ensures that transactions are secure, scalable, and resistant to tampering, making it ideal for real-world applications in decentralized finance, gaming, and enterprise systems.

The 2PC-MPC protocol represents a significant advancement in cryptographic security for decentralized systems. Its integration with Mysticeti's broadcast communication framework ensures that the protocol is not only secure but also highly scalable and efficient. This combination of cryptographic rigor and practical performance positions Ika as a leader in decentralized infrastructure, setting the standard for secure and scalable blockchain operations.

3.3 Controlling dWallets with Smart Contracts

The integration of dWallets with smart contracts unlocks powerful capabilities for developers, enabling programmable, secure, and decentralized applications. dWallets, powered by the 2PC-MPC protocol and made available by Ika, act as programmable building blocks that interact seamlessly with smart contracts to enforce logic, manage assets, and execute complex workflows. In particular, platforms like Sui, with its parallel transaction execution and Move-based smart contracts, provide an ideal environment for leveraging the advanced features of dWallets.

3.3.1 Programmable Control and Interoperability

dWallets are inherently programmable, allowing their behavior to be defined and governed by external smart contracts. This programmability transforms dWallets into active components of decentralized applications, capable of enforcing rules and executing transactions based on predefined conditions.

For example, a Move-based smart contract on Sui can instruct a dWallet to execute a payment only when certain criteria are met, such as multi-user approval or specific on-chain events. Similarly, a DeFi protocol can use dWallets to automate liquidity management, where the wallet responds to market conditions or governance decisions to deposit or withdraw assets. This interplay between dWallets and smart contracts enables complex logic to be executed securely and trustlessly, expanding the design space for decentralized applications.

3.3.2 Technical Workflow of dWallet-Smart Contract Interaction

When a dWallet is bound to a smart contract, in order for a transaction to be signed by that dWallet, it must first be approved by that smart contract. Ika enforces that by verifying a state proof of the smart contract state.

For instance, in Sui, a dWallet controlling a Bitcoin account with 10 BTC is bound to a Move smart contract implementing a DAO treasury that requires approval by a majority of stakeholders to move funds. In order for the Bitcoin signature to be generated by the dWallet, Ika will first verify that the DAO's smart contract state on Sui contains the transaction in the approved list of transactions.

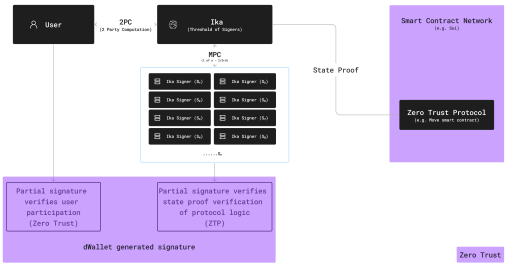

Figure 4: Sui Smart Contract Controlling a dWallet
Ika verifies a Sui state proof to determine whether the Sui smart contract approved a transaction, and the dWallet signature is cryptographic proof both the user and Ika participated.

By integrating dWallets with smart contracts, Ika enables developers to create decentralized applications that are both programmable and secure. This interaction empowers dWallets to act as autonomous agents, executing complex workflows defined by smart contracts while upholding the highest standards of cryptographic security. The combination of dWallets, the 2PC-MPC protocol, and platforms like Sui sets the stage for a new generation of decentralized applications, where logic, security, and scalability converge seamlessly.

4 Ika Security

4.1 Sui Security

Ika Network builds on a fork of Sui's codebase[13], inheriting not only its consensus mechanism but also its broader architectural and security assumptions. Sui's design[14] integrates several innovative elements, including the Mysticeti consensus algorithm, the Move programming language[15], and a modular blockchain architecture. These components collectively ensure scalability, performance, and security in a distributed environment.

Sui's security is grounded in Mysticeti, a Byzantine Fault Tolerant (BFT) protocol optimized for high throughput and low latency. Mysticeti employs a Directed Acyclic Graph (DAG)-based approach to ensure reliable broadcast communication, enabling consistent transaction ordering and data availability. The protocol assumes that a threshold of validators by stake behave honestly and follow the protocol. Under this assumption, Mysticeti guarantees both safety—ensuring no two honest validators observe conflicting states—and liveness, even under network delays.

Beyond consensus, the Sui codebase incorporates robust security practices to minimize vulnerabilities. The Move programming language emphasizes safety and formal verification. Move is designed to prevent common classes of vulnerabilities, such as reentrancy attacks or unauthorized state modifications, through its resource-oriented programming paradigm. Move enforces strict rules around asset ownership and data manipulation, reducing the risk of unintended behaviors or exploits.

Ika inherits these properties, making the network secure under the same assumptions. Any vulnerabilities in Sui's architecture, consensus mechanism, or codebase could also affect Ika, highlighting the importance of rigorous testing and continuous improvement. The integration of Mysticeti ensures high performance for the 2PC-MPC protocol and ties Ika's security to the soundness of the Sui framework as a whole.

4.2 Blockchain / Economic Security

As a Proof of Stake (PoS) system, Ika inherits the general security assumptions and risks associated with PoS blockchains such as Denial of Service (DoS) attacks, long-range attacks, or malicious collusion among signers (51% attack). Economic security in the Ika network is based on the assumption that signers stake tokens, thereby aligning their incentives with honest behavior. The network's security is further assumed to be economically viable under the condition that it achieves sufficient transaction volume to maintain self-sufficiency and incentivize signers to participate in the MPC protocols and the consensus (or block production). If this assumption is not met, the network's security and relevance may be

undermined, potentially necessitating adjustments to its operational parameters. These adjustments could include changes to monetary policy, fee structures, consensus rewards, governance frameworks, or technical components such as computation requirements or intervals.

Governance in the Ika network contributes to its economic security under the assumption that decision-making processes and frameworks effectively manage network modifications and protocol updates. This assumes efficient decision-making, timely issue resolution, and consensus among stakeholders to avoid network splits, which could compromise the network's stability and reliability. The security of the governance model also assumes that no single group of stakeholders will exert disproportionate influence, ensuring a decentralized power dynamic that reflects the broader community's interests and resists exploitation for malicious purposes.

The security of Ika's 2PC-MPC protocol relies on the assumption that a threshold of the signers by stake remains honest, active, and compliant with the protocol. Unlike traditional blockchains, where forks may serve as a fallback mechanism, Ika assumes a stable threshold of honest signers since its homomorphic decryption key is collectively held by the signers, and ceases to function if less than the threshold participates honestly. In a Zero Trust setup, the protocol assumes that no group with over a third by stake of malicious signers will refuse participation, to avoid potential censorship or ransom scenarios. In a non-Zero Trust setup, security assumes that no collusion exceeding the threshold of the signers by stake will occur, to avoid direct compromise of user assets.

4.3 Cryptographic / Code Security

Ika's security relies heavily on advanced cryptographic techniques, including 2PC-MPC, threshold homomorphic encryption[16][17], and various digital signature algorithms such as ECDSA[18], EdDSA[19], and Schnorr[20]. The security of these systems is based on the soundness of the underlying cryptographic primitives and protocols as defined in their respective research literature.

The cryptographic security assumption depends on the implementation, and the absence of bugs, side-channel attacks, or weaknesses in third-party dependencies, cryptographic libraries, or other parts of Ika's source code that can introduce vulnerabilities, failures or data corruption that undermine the system. For instance, a flaw in the implementation of threshold encryption could expose secret materials, compromising the integrity of dWallets. Additionally, cryptographic advances or breakthroughs (e.g., in quantum computing) could impact the long-term security assumptions.

4.4 Smart Contract Security

Smart contracts governing dWallets are another critical component of Ika's security model. Vulnerabilities in the smart contract code, the underlying blockchain network, or the mechanisms for verifying state proofs can result in unintended behavior or security breaches. For example, a bug in a Sui-based smart contract interacting with a dWallet could lead to unauthorized transactions or loss of funds.

Users interacting with Ika through smart contracts, must also inherit the security assumptions of those contracts, of the smart contract network (e.g. Sui), and the quorum-based smart contract network light client / state proof and its implementation on Ika and other blockchain risks such as forks, consensus failures, DoS etc. resulting from that interaction. This layered dependency adds complexity to the overall security assessment and highlights the importance of robust contract development and auditing practices.

4.5 Signer Security

Ika signers operate the nodes that are responsible for the collaborative signing process in the 2PC-MPC protocol, and its security assumes they remain operational and secure from infrastructure breaches, malware, or network disruptions. This assumption applies even if the operators of these nodes act honestly, as such attacks could still compromise the system. Additionally, as signers can change between epochs, a fixed set of signers trusted by the user is not a security assumption of Ika.

Signers also have the potential to exploit their privileged position to engage in strategies like front-running or Maximal Extractable Value (MEV) extraction. Such behavior, while not necessarily malicious in the traditional sense, can undermine the fairness and integrity of the system.

4.6 End-User Security

The security of Ika and its dWallets assumes that end-users can securely manage their private keys, user shares, and authentication methods. This includes protecting against phishing attacks, compromised software or hardware, and other exploits targeting these critical components. The security model presumes that users safeguard their share of the

dWallet and any associated decryption or authentication keys. If this assumption is not upheld, the compromise of these elements could allow attackers to gain unauthorized access over the associated dWallet, leading to potential loss of funds or control.

This section provides a high-level overview of the security assumptions, risks, and potential adverse scenarios associated with Ika. While the system incorporates advanced cryptographic techniques and state-of-the-art consensus mechanisms, its security fundamentally depends on the honest participation of signers, the integrity of its cryptographic primitives, and the resilience of its implementation and user practices.

5 Use Cases

Ika's dWallet framework and the 2PC-MPC protocol provide a robust foundation for diverse applications across blockchain ecosystems. By combining cryptographic security with programmability and high performance, Ika offers a versatile solution for decentralized finance (DeFi), gaming, institutional use cases, and more. This section explores the practical implications of Ika's architecture, drawing on both its Zero Trust principles and its ability to deliver high throughput and low latency.

5.1 Decentralized Finance (DeFi)

In the DeFi ecosystem, security and performance are critical. Existing solutions often rely on trust in intermediaries or centralized systems to manage liquidity, execute trades, and transfer assets across chains. These approaches introduce vulnerabilities and inefficiencies, such as delays in transaction execution or the risk of exploits targeting centralized points of failure.

Ika's dWallets address these challenges by enabling secure and programmatic control over assets. For example, a DeFi protocol could use a dWallet to automate liquidity management, rebalancing funds across pools based on real-time market conditions. The protocol's logic, encoded in a smart contract, would interact directly with the dWallet, ensuring that transactions are executed securely and with minimal latency. With native interoperability, these operations avoid the risks associated with wrapped tokens or bridging solutions, providing users with a seamless and efficient experience.

The high throughput of the 2PC-MPC protocol ensures that dWallets can handle the rapid transaction demands typical in DeFi, such as high-frequency trading or collateral liquidation. This combination of security, performance, and programmability positions Ika as a foundational layer for next-generation DeFi applications.

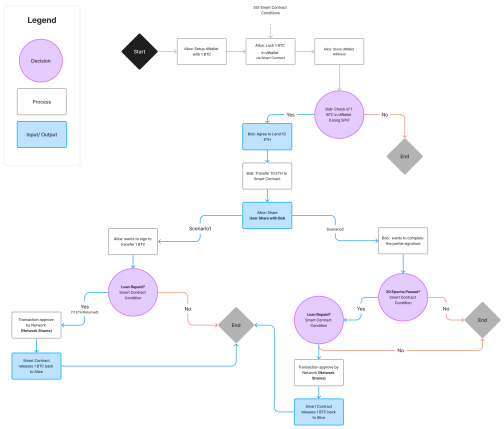

Figure 5: Multi-Chain Lending Orchestrated from Sui

Alice uses her native BTC as collateral for a native ETH loan through a smart contract on Sui. Zero Trust Architecture is enforced using dWallets.

5.2 Gaming and NFTs

The gaming industry and NFT marketplaces demand secure, scalable solutions to manage in-game economies and digital collectibles. Current systems often struggle with performance bottlenecks, particularly during high-demand periods such as NFT drops or in-game asset trading.

dWallets enable a secure and efficient framework for these applications. In a gaming ecosystem, a dWallet could manage player-owned assets, enforcing transfer rules or approving trades based on in-game logic. For example, a blockchain-based game might use a dWallet to handle cross-platform asset use, allowing players to utilize their digital items seamlessly across games or ecosystems without relying on 3rd party intermediaries.

NFT marketplaces can leverage dWallets to implement programmable ownership and transfer rules and even transfer of an entire multi-chain NFT portfolio. A dWallet could ensure that high-value NFTs are transferred only under specific conditions, such as multi-party approval or after a defined holding period. The protocol's high performance ensures that

transactions are processed without delays, even during peak activity, enhancing the user experience and supporting scalable market operations.

5.3 Institutional and Enterprise Solutions

Institutions managing digital assets require systems that balance security, compliance, and performance. Traditional custodial solutions often force institutions to compromise on decentralization, while existing decentralized alternatives lack the programmability, performance and security needed for enterprise-scale operations.

Ika's dWallets provide a decentralized yet programmable alternative. A corporate treasury could use a dWallet to automate fund disbursements, enforce multi-signature approval for high-value transactions, and maintain full audibility of all operations. For instance, a multinational company could integrate dWallets with its payment systems to manage cross-border transfers securely and efficiently, adhering to regulatory requirements while maintaining operational flexibility.

The integration of the 2PC-MPC protocol ensures that these operations enforce Zero Trurst security, while Mysticeti's high-performance broadcast channel guarantees low-latency processing. This combination enables enterprises to scale their blockchain operations without compromising security or performance.

5.4 Decentralized Autonomous Organizations (DAOs)

DAOs require transparent and secure mechanisms for managing multi-chain treasuries and executing governance decisions. Traditional solutions often rely on centralized intermediaries for multi-chain treasury management, undermining the decentralized ethos of DAOs.

With Ika's dWallets, DAOs can enforce governance rules programmatically. A DAO could use a dWallet to execute spending decisions on any chain following a community-approved proposals. For example, a treasury-managed dWallet could release BTC only after achieving a quorum of votes, with the signing process distributed securely across the network. The high performance of the 2PC-MPC protocol ensures that these transactions are executed quickly, enabling DAOs to respond efficiently to changing conditions.

The programmability and transferability of dWallets offer DAOs capabilities that closely mimic real-world organizational structures, such as companies with subsidiaries or joint ventures. With dWallets, DAOs can implement hierarchical control mechanisms where a parent DAO delegates specific powers to subsidiary DAOs, each controlling their own dWallets. These subsidiaries can operate semi-independently while remaining cryptographically tied to the parent DAO's governance structure.

Furthermore, the transferability of dWallets allows DAOs to treat their assets and operations as modular components that can be sold, transferred, or merged, similar to mergers and acquisitions in the corporate world. For instance, a DAO could spin off a particular project or sub-organization by transferring its associated dWallet, complete with all its underlying assets and logic, to a new set of owners or operators. This flexibility provides a practical pathway for scaling decentralized governance structures while maintaining transparency and security.

5.5 Multi-Chain Atomic Swaps

Ika Network enables secure and trustless multi-chain atomic swaps, facilitating seamless exchanges of native assets across different blockchains. Unlike traditional approaches that often rely on intermediaries or wrapped tokens, Ika's architecture ensures that swaps are executed natively, preserving the integrity and security of the participating blockchains.

In a multi-chain atomic swap, two dWallets collaborate to enable a direct exchange of assets. Each dWallet manages the native asset on its respective blockchain, and the transaction logic ensures that both sides of the swap occur simultaneously. This atomicity guarantees that either both transfers complete or neither does, eliminating the risks associated with partial or failed transactions.

For instance, consider a swap between Bitcoin (BTC) and Ethereum (ETH). A dWallet on the Bitcoin network securely handles the BTC transfer, while another dWallet on Ethereum facilitates the ETH transfer. A smart contract on Sui ensures that the signing processes for both transactions are linked, so neither asset can be transferred unless the other transfer is guaranteed to occur. This coordination maintains the security guarantees of the respective blockchains while enabling direct asset swaps without intermediaries.

5.6 Retail User Custody

For individual users, dWallets provide a unique combination of security, flexibility, and control. Unlike traditional wallets, which rely on a single private key managed by the user or a custodian, dWallets dis-

tribute signing responsibility across the user and the network. This design protects users from the risks of key loss or theft while enabling advanced features such as programmable recovery mechanisms.

For example, a user could configure their dWallet to execute recurring payments or limit daily spending automatically, ensuring both convenience and security. In the event of losing the authentication key that holds the permissions for the dWallet, the dWallet could implement a social recovery mechanism that reassigns ownership after a predefined waiting period, minimizing the risk of permanent asset loss.

With native interoperability, users can manage assets across multiple blockchains from a single application utilizing dWallets, avoiding the complexity and risks associated with intermediary services. The high performance of the 2PC-MPC protocol ensures that these operations remain seamless, even during high-demand periods.

Ika's dWallets and the 2PC-MPC protocol address critical limitations in existing blockchain infrastructure, offering a secure, high-performance, and programmable foundation for decentralized applications. From DeFi and gaming to enterprise and cross-chain operations, dWallets enable use cases that are not only secure but also scalable and efficient. By integrating cryptographic rigor with developer-friendly programmability, Ika provides a robust framework for the next generation of blockchain systems.

6 Economic and Governance Model

The Ika economy is designed to align incentives, ensure network security, and enable decentralized decision-making. It revolves around the utility of the native IKA token, which supports staking, transaction operations, and governance.

Gas Fees: Gas fees, denominated in IKA tokens, are charged for network operations such as creating a new dWallet via Distributed Key Generation (DKG) or signing transactions with a dWallet. These fees reward signers for their contributions and help prevent spam and denial-of-service attacks.

Proof-of-Stake Mechanism: The network's proof-of-stake mechanism is used to select and incentivize signers and delegators who operate and secure the Ika platform. Signers stake IKA tokens to participate in the 2PC-MPC protocol, while delegators support signers and share in the staking rewards e.g. gas fees, inflation.

On-Chain Governance: Ika employs a decentralized on-chain governance model of IKA stakers/holders, allowing proposals on protocol upgrades, economic adjustments, and other changes to the network - according to the then relevant mechanism defined by the protocol. This ensures that the protocol evolves transparently and aligns with the interests of its participants.

This framework ensures that Ika remains secure, scalable, and adaptable, supporting its role as a foundational layer for decentralized applications.

7 Conclusion

Ika represents a significant advancement in blockchain interoperability, security, and scalability. By combining the decentralized architecture of dWallets, the cryptographic rigor of the 2PC-MPC protocol, and the high performance enabled by the Mysticeti consensus protocol, Ika delivers a secure and efficient foundation for next-generation decentralized applications.

The network's Zero Trust framework eliminates reliance on trusted intermediaries, ensuring that all operations are approved by the user. Its high-performance design supports low-latency, high-throughput operations, making it suitable for a wide range of use cases, including decentralized finance, gaming, institutional asset management, and cross-chain interoperability. Programmable dWallets, integrated seamlessly with smart contracts, act as building blocks for developers to create complex and dynamic workflows that were previously infeasible.

The IKA token economy aligns incentives, secures the network, and empowers participants through decentralized governance. By balancing cryptographic innovation, economic sustainability, and community-driven development, Ika provides a robust platform for the decentralized economy's growth.

As the blockchain ecosystem continues to evolve, Ika stands ready to address its most pressing challenges, offering a future where security, scalability, and decentralization coexist without compromise. We invite developers, enterprises, and ecosystem participants to join us in building on this transformative infrastructure and shaping the next chapter of decentralized innovation.

References

[1] Babu Pillai, Zhé Hóu, Kamanashis Biswas, Vinh Bui, and Vallipuram Muthukkumarasamy. Blockchain interoperability: performance and

security trade-offs. In *Proceedings of the 20th ACM Conference on Embedded Networked Sensor Systems*, pages 1196–1201, 2022.

[2] Giulio Caldarelli. Wrapping trust for interoperability: A preliminary study of wrapped tokens. *Information*, 13(1):6, 2021.

[3] Xuan-Thao Nguyen and Jeffrey A Maine. Crypto losses. *University of Illinois Law Review*, 2024(4), 2024.

[4] Kunpeng Ren, Nhut-Minh Ho, Dumitrel Loghin, Thanh-Toan Nguyen, Beng Chin Ooi, Quang-Trung Ta, and Feida Zhu. Interoperability in blockchain: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(12):12750–12769, 2023.

[5] Omer Sadika, David Lachmish, Yehonatan C Scaly, and Sheran Hussain. Ika: Zero trust mpc network. White paper to be published.

[6] Offir Friedman, Avichai Marmor, Dolev Mutzari, Omer Sadika, Yehonatan C Scaly, Yuval Spiizer, and Avishay Yanai. 2pc-mpc: Emulating two party ecdsa in large-scale mpc. *Cryptology ePrint Archive*, 2024.

[7] Kushal Babel, Andrey Chursin, George Danezis, Anastasios Kichidis, Lefteris Kokoris-Kogias, Arun Koshy, Alberto Sonnino, and Mingwei Tian. Mysticeti: Reaching the limits of latency with uncertified dags. *arXiv preprint arXiv:2310.14821*, 2023.

[8] Jens Groth and Victor Shoup. On the security of ecdsa with additive key derivation and presignatures. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 365–396. Springer, 2022.

[9] Offir Friedman, Avichai Marmor, Dolev Mutzari, Yehonatan C Scaly, and Yuval Spiizer. Practical zero-trust threshold signatures in large-scale dynamic asynchronous networks. This paper, describing the new version of the 2PC-MPC protocol including some of the aspects discussed in this white paper, will be published soon.

[10] George Danezis, Lefteris Kokoris-Kogias, Alberto Sonnino, and Alexander Spiegelman. Narwhal and tusk: a dag-based mempool and efficient bft consensus. In *Proceedings of the Seventeenth European Conference on Computer Systems*, pages 34–50, 2022.

[11] Alexander Spiegelman, Neil Giridharan, Alberto Sonnino, and Lefteris Kokoris-Kogias. Bullshark: Dag bft protocols made practical. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 2705–2718, 2022.

[12] George Danezis and David Hrycyszyn. Blockmania: from block dags to consensus. *arXiv preprint arXiv:1809.01620*, 2018.

[13] Sui. https://github.com/MystenLabs/sui.

[14] Sam Blackshear, Andrey Chursin, George Danezis, Anastasios Kichidis, Lefteris Kokoris-Kogias, Xun Li, Mark Logan, Ashok Menon, Todd Nowacki, Alberto Sonnino, et al. Sui lutris: A blockchain combining broadcast and consensus. *arXiv preprint arXiv:2310.18042*, 2023.

[15] Sam Blackshear, Evan Cheng, David L Dill, Victor Gao, Ben Maurer, Todd Nowacki, Alistair Pott, Shaz Qadeer, Dario Russi Rain, Stephane Sezer, et al. Move: A language with programmable resources. *Libra Assoc*, page 1, 2019.

[16] Offir Friedman, Avichai Marmor, Dolev Mutzari, Yehonatan C Scaly, Yuval Spiizer, and Avishay Yanai. Tiresias: Large scale, maliciously secure threshold paillier. *Cryptology ePrint Archive*, 2023.

[17] Lennart Braun, Ivan Damgård, and Claudio Orlandi. Secure multiparty computation from threshold encryption based on class groups. In *Annual International Cryptology Conference*, pages 613–645. Springer, 2023.

[18] Don Johnson, Alfred Menezes, and Scott Vanstone. The elliptic curve digital signature algorithm (ecdsa). *International journal of information security*, 1:36–63, 2001.

[19] Simon Josefsson and Ilari Liusvaara. Edwards-Curve Digital Signature Algorithm (EdDSA). RFC 8032, January 2017.

[20] Claus-Peter Schnorr. Efficient signature generation by smart cards. *Journal of cryptology*, 4:161–174, 1991.