Security Challenges in Building Blockchains Bridges and Countermeasures

Sai Pranav Krishna
Liverpool John Moores University

Singh, Pushpa
Deptt. Of Computer Science & Engg., GL Bajaj Institute of Technology & Management

https://doi.org/10.5109/7151703

出版情報：Evergreen. 10 (3), pp.1558-1569, 2023-09. 九州大学グリーンテクノロジー研究教育センター
バージョン：

Security Challenges in Building Blockchains Bridges and Countermeasures

Sai Pranav Krishna[1], Pushpa Singh[2*]

[1]Liverpool John Moores University

[2]Deptt. Of Computer Science & Engg., GL Bajaj Institute of Technology & Management, Greater Noida

*Author to whom correspondence should be addressed:

E-mail: pushpa.gla@gmail.com

(Received April 22, 2023; Revised July 8, 2023; accepted July 20, 2023).

Abstract: The implementation of distributed ledger technologies and Blockchains has recently become high in various domains, from finance to governance. There are multiple Blockchain frameworks and networks for different use cases but still seems to be a gap in how different frameworks, protocols, and ledgers interact with each other. With the increase in the adoption of Blockchains, there is an increasing need for suitable interoperable solutions so that more value can be provided to the end user. This research focuses on analyzing how current Blockchain bridges are built and evaluating common security risks and countermeasures within the scope of interoperability. A threat model is proposed to analyze blockchain interoperability's various components, vulnerabilities, risks, and corresponding mitigation techniques. Common security vulnerabilities like centralization of trust and vulnerable smart contracts and others were identified and classified based on the kind of bridge component along with possible mitigations. Each solution, like Relays, HTLCs, Notary Schemes, and Smart Contracts, is explored. Thus, the study will help developers understand the risks involved by providing insights and pointing out the need for standardization on Blockchain interoperable solutions.

Keywords: Blockchain; Interoperability; security; Relays; Notary Schemes; Smart Contract

1. Introduction

Blockchain is a decentralized and distributed digital ledger technology that permits multiple participants to maintain a shared database without relying on a central authority. It was initially introduced as the underlying technology for cryptocurrencies like Bitcoin, but its applications have expanded to various industries. Blockchain technology has been exploited in various areas such as supply chain management (SCM), healthcare, finance, voting systems, identity verification, and more. The adoption of Blockchain is inevitable, but barriers and risks remain[1)] and one of the major risks is interoperability and system integration. Interoperability in blockchains is how Blockchains can talk to each other or other systems. Interoperability in Blockchains is the ability of one ledger to invoke certain events in another Blockchain or react to certain transactions happening on another Blockchain. This operation, on the whole, is called interoperation[2)]. This can be achieved in multiple ways. However, it can be broadly classified into Bridges, Direct, Connectors, and others[3)]. Each of these will be used in different scenarios and application requirements. The key point in these protocols is to have that balance between the need for interoperability and maintaining the spirit of decentralization. While the above protocol-level patterns help communicate between two or more ledgers, ledger-level features such as locking, burning, and wrapping of assets will be used to achieve interoperability. Locking of assets can be defined as moving assets to a place in the source Blockchain not being usable, as they can be minted in another network. Wrapping of an asset can be defined as a re-creation of the same asset in the recipient Blockchain, which will theoretically have the same value as it is in the source Blockchain. Burning is another mechanism where the asset is destroyed in the source Blockchain so that an equal amount of assets can be recreated in other Blockchains. Typical problems with achieving interoperability include having atomicity in interoperability transactions, the medium of communication between the Blockchains, such as bridges not being truly decentralized, and smart contract vulnerabilities.

Even with the advent of many Blockchain frameworks, no network fits all the requirements[4)]. Thus, there is a need to create a way for Blockchains to talk to each other. However, because of the characteristics of the technology, they are not built to interact with each other [5)]. And even with current solutions such as bridges and other

mechanisms, data exchange heavily depends on the credibility of such integration[6]. With the total market cap of Blockchain assets touching $3 trillion as of November 2021[7], and the demand for asset transfer across multiple chains rising, there is a need for defining common security risks associated with such bridges between Blockchains and countermeasures. Even with multiple frameworks and protocols being available, the knowledge necessary for achieving interoperability is still fragmented[8]. The recent hack on the wormhole Blockchain bridge, which is used to transfer assets between Ethereum and Solana networks, where the exploiter was able to steal assets worth $320 million because of the bridge contract vulnerability[9], making risks involved in bridges huge. Reference[10] has represented a bar diagram to understand the recently stolen funds. As "Interoperability is key to Survivability" [11], so, it is crucial that these interoperable solutions, especially the bridges, maintain many security standards. Blockchain interoperability is a term that can mean any of the following:

a. Interoperability between Blockchain and an existing legacy system
b. Interaction between two chains
c. Interoperability between two smart contracts[12].

The scope of this study is limited to interoperability between two chains and examining current interoperability frameworks regarding security risks and vulnerabilities involved. Threat model analysis helps to find and analyze the system's complexity and possible attacks. Priority-based threats are identified, which will help us understand the intensity and possible mitigations. In this study, a case flow is considered, and then with 6 step analysis, threats are identified and ranked. Hence, this research aims to analyze the Blockchain bridges and security risks among current Blockchain bridge patterns and propose countermeasures. Below are the objectives of this research paper:

- Provide a concise review of the current Blockchain interoperability requirements and needs, which should help better understand the risks that can be taken along.
- Provide threat analysis and analyze Blockchain bridge design patterns to assess the security risks.
- Propose measures to mitigate the security risks associated with Blockchain bridges.
- Analyzing the limitations and risks with proposed countermeasures.

2. Literature Review

Blockchain has been proven as a critical technology in view of existing pandemic control to offer a reliable, efficient, and low-cost means of better decision-making[13].

Blockchain means a network of parties or nodes that maintain a shared state in a specific order to form a ledger. Trust in the network is ensured by the diversity of the group and their number. A transaction can propose a change in this ledger, or the change itself can be tracked by the transaction based on network design. These transactions are usually aggregated into blocks and linked to the previous block's hash, thus called Blockchains. Bitcoin was the first Blockchain to transfer value across a cryptocurrency, also known as Bitcoin[14]. Recent Blockchains have programmable capabilities, which means their state machine is extended with an application written by users. These programs are called smart contracts. Ethereum is one such prominent network that uses EVM (Ethereum Virtual Machine) to execute the programs.

The global state of the ledger or order in which these transactions should be added to the ledger is determined by the consensus algorithm that the ledger employs. These ledgers' main purpose would usually be collaboration or privacy[15]. Consensus algorithms define how nodes interact with each other and how the global state evolves[16]. Some common algorithms include Proof-of-Work (PoW), where nodes have to solve a cryptographic puzzle to validate transactions. Proof-of-Stake (PoS) is another commonly used algorithm where an entity's stake determines who can validate transactions. On the other hand, there are other algorithms like proof of burn (PoB), where validators can mine the transactions after sending a certain amount of value to an unreachable address, and Proof of Elapsed Time (PoET), where nodes have a certain waiting time before they validate. PoET is commonly used in private Blockchains[17].

2.1 Interoperability and Cross-Chain Communication

Interoperability is defined as the ability of software systems to exchange and use information. Vernadat defined interoperability as a scope where two or more systems can provide or accept service from other systems and use the common exchange of data accurately[18]. Interoperability can also be defined as the distributed ledger systems' ability to process transactions originating in another distributed ledger with homogenous or heterogenous identity management, cryptographic management, consensus mechanism, and smart contracts capabilities[19].

Cross-chain Communication refers to a procedure where two chains interact with each other in order to have consistent and in-sync data. It can be considered as Blockchains of a different nature communicating with each other at an interchain level. One such existing implementation is the Inter ledger protocol which enables two chains to transfer value across[20]. Due to the variety of designs and operational nature of each Blockchain, it is not straightforward to create a cross-chain communication protocol. One such major hurdle is when one chain wants to verify the change record in another. As per the cross-Blockchain proof problem[12], It is hard, if not impossible,

to detect and verify data recorded on one chain just by observing the exchanged information from another chain, meaning it can be hard for the target Blockchain in communication to fully detect the change that is intended and to verify the transaction requiring the change. Thus, it is impossible for Blockchains to talk to each other directly[8]. This is where trusted third-party solutions come in helpful for transferring data from one chain to another. But such third-party integration or solutions is against one of the Blockchain features of decentralization and have no single point of failure. Thus the interoperability framework should be able to let ledgers transfer information and value in a trustworthy manner. Authors in[6] mentioned that cross-Blockchain transactions might not make changes to the state directly; instead, they trigger some set of functionalities on the other chain, which may change the state there.

2.2 Types of available interoperable solutions

A wide range of interoperable solutions is currently available; while some are deployed and in use, most are still part of the literature. Types of which Blockchain interoperable solutions are part can be classified based on either integration type, consensus mechanism, level of decentralization, and design rationale. Buterin, in his chain interoperability[4] classified the solutions into three types:

a) Centralized or multi-sig Notary Schemes, where a group of parties agrees to act on a destination chain on a trigger from the source chain.

b) Side Chains / Relays which are systems of one Blockchain that can read others.

c) Hash-locking is a mechanism in which two chains communicate with each other based on a trigger revealing a hash.

In reference[21], the authors discuss Blockchain of Blockchains (BoB) and Hybrid Connectors. Further, each category can be divided into sub-categories like Side chains having different mechanisms like a centralized two-way peg, federated two-way peg, and simplified two-way payment verification proofs. A bridge solution can contain multiple solutions working together to achieve cross-chain communication. Several solutions provided in the study are shown in Table 1.

Table 1. Solutions offered across multiple studies.

Reference	Solution Category
3)	Transaction and Smart Contract (SC)-based
4)	Notary Schemes, HashLocked Contracts
8)	HashLock Contracts Public Connectors
9)	Improved Hash Locking
11)	Blockchain Gateways, Bridges and Delegated Hash-Locks
15)	Explored principles and schemes to achieve practically interoperable. Blockchains and comparison on existing solutions.
21)	Blockchain Of Blockchains, Hybrid Connectors
22)	Public and Hybrid connectors
23)	Atomic Swaps
24)	Public and Hybrid connectors
25)	Relay Schemes, Hash Locking
26)	Public Connectors
27)	Side chains and two-way pegs. Analyses state of art side chains and their limitations and remedies
28)	Notary Schemes, Hybrid Solutions
29)	Public Connectors, Blockchain of Blockchains
30)	Proposes a model based on hash locks and multi-sig notaries
31)	Proposes a new model for permissioned Blockchains based on pub-sub.
32)	Proposes a trustless bridge to transfer assets from BFT Blockchain to other chains that support smart contract execution
33)	Blockchain Oracle Voting Based
34)	Conditional Transaction Based
35)	Smart Contract and Wrapped Tokens
36)	Propose an application-based cross-chain interoperability solution named appXchain which allows Blockchain networks of any architecture type and industrial focus to inter-communicate
37)	Analyses state of art side chains and their limitations and remedies. Defined a fully verifiable interaction model.

[38]	Analyses many security and privacy issues for blockchain interoperability methods like Notary Schemes, Sidechains and Hashed Time-Lock Contracts along with their possible mitigation strategies.

From Table 1, we can observe various solutions to achieve interoperability, and each of the use cases is developed with different requirements. Bridge is one of the way to achieve blockchain interoperability, so that they can link and transfer the data and digital assets[11]. By using a notary, cross-chain transactions can be validated and notarized to guarantee their integrity and correctness[4,28,38]. For example, transaction-based solutions[3,34] primarily focus on cross-chain communication using transactions that can be submitted independently without a framework. As discussed in [4,8,9,11,25,30,32,38], the hash-locking mechanism is at a protocol layer to achieve an atomic transfer of value. Sidechains can act as intermediaries for transferring assets or data between different blockchains [27,37,38].Blockchain of Blockchain[21,29] public connectors [8,22,24,26,29] and hybrid connectors[21,22] permits various level of interoperability between blockchains and other distributed ledger technologies. In Table 2, we will compare the observations made in some publications.

Table 2. Observations made in different studies.

Reference	Observation
[22]	This Survey focuses more on Sidechains, routers, and Smart contracts. A brief introduction to industry solutions is also done.
[23]	Quantitative analysis on the fairness of atomic swaps and proposes a new fair swap protocol
[24]	Discussed Industry side chain solutions like Plasma, Polkadot, TAST, and Metronome
[25]	Describes a decision scheme to decide on optimal solution suits for interoperability
[26]	Explains cross-chain communication standards and reduce them into a fair exchange problem
[27]	Extensive examination of side chains with pros and cons
[33]	Proposed Blockchain interoperability oracle for voting-based approach
[35]	Summarizes wrapped tokens and issuing procedures.

Even several microfinance managers considered blockchain irrelevant[39] to the microfinance sector may be due to its interoperability challenges. Despite Blockchain interoperability seeming to be complicated to achieve as each of them is built in different ways and there seems to be no single solution that suits all the needs, it seems it is possible to have Blockchains connected with each other. But looking forward, "there is a scant effort today to address the standardization of the various infrastructure building blocks, messages, data formats, and flow to support the interoperability across Blockchains" [11]. Even if the interoperable protocols are different and shine in the different use cases of interoperability, there can be a common framework for the data flow. The current frameworks in use, as in Fig. 1, include side chains, relays, the Blockchain of Blockchains and smart bridge contracts. It is important to note that these solutions can complement each other and work together to achieve interoperability. The relay component here transmits data from one chain to another. It fetches data from one chain and posts it into the smart contract on another chain that would keep track of the host chain. On the other chain, this relay contract can help validate the proofs submitted by the client, which can be a system or user. The smart bridge contract on the chain facilitates the locking, unlocking or burning of tokens or data handling. All these can work together; for example, relays are often attached to a smart contract on one or both chains, which will verify the cross-chain transaction from then. It is also observed how private or consortium Blockchains need a different interoperable model while considering their properties of restricted entry into the chain or limited participation in mining or validation of the blocks. It is important to note that these solutions sometimes can work together, Like smart contracts and bridges. Hashlock contracts and conditional transactions.

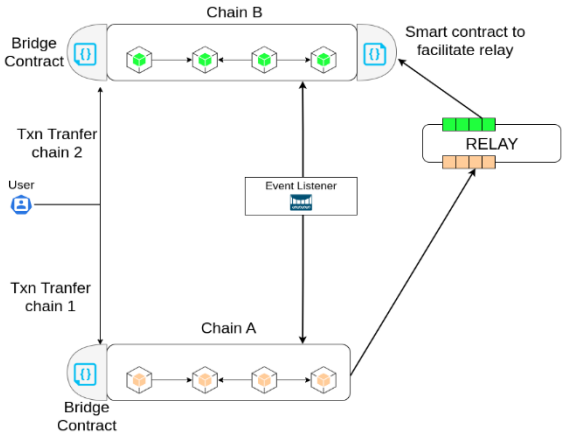

Fig. 1: Various ways for chains to communicate

3. Research Methodology

This paper mainly presents the contribution of Blockchain technology in considering the wide range of

Blockchain chain protocols and the scenarios they fit in and design aspects; there came different interoperable frameworks and design approaches. The proposed threat model is used to analyze the interoperability security risk of Blockchain. If any vulnerabilities find, rank them and their corresponding adverse effect. Now find out the suitable technique to mitigate identified risk. Overall, the methodology of the proposed threat model is shown in Fig. 2. First of all, collect data regarding system such as components, processes and interaction between component. After that each component have analyzed and categorize the type of vulnerability. Then rank the vulnerable object and its corresponding adverse effect. Step 4 review the associated risks and suitable action to mitigate the risk. Step 6 documented the findings of step 5. Review the system again with added security implementation to check any further risks are identified or not. In case of possibility of risks need to collect data again and repeat the process.

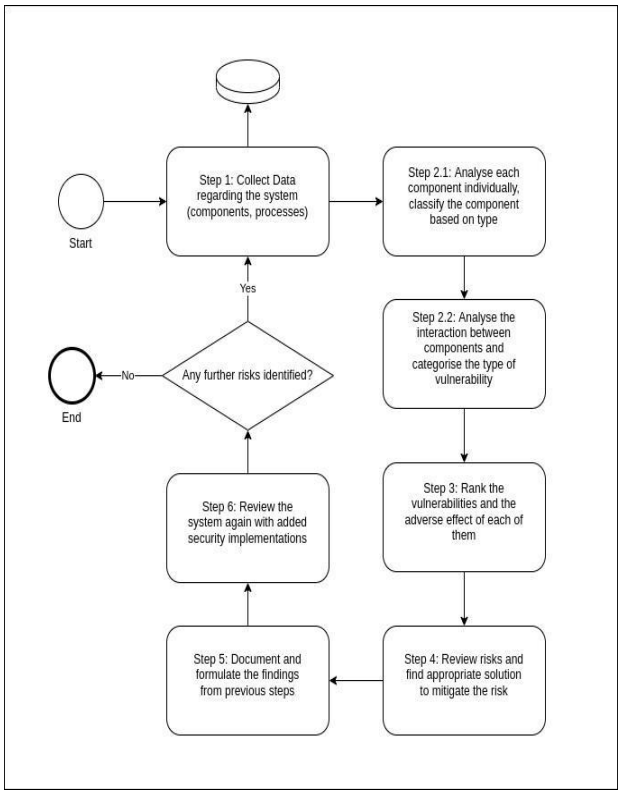

Fig. 2: Steps to conduct a threat model

3.1 Interpretation and Analysis

The coding rules proposed by[39] were followed to analyze the data retrieved in the previous step. First, all the articles were read carefully to analyze the artefacts and identify corresponding characteristics and patterns in the sense of Blockchain interoperability and bridges. Crucial details like title, author and frameworks covered were noted along with observations. The list was iterated again to avoid qualitative errors (different terminology meaning the same thing). Once the basic understanding of the current Blockchain interoperability was performed, the study of security aspects of current Blockchain systems and advanced interoperability risks were studied. Artefacts identified similarly are grouped and compared to further analyze the pattern. A quantitative approach with a case study analysis method is being used in this study. To better assess the risks and vulnerabilities in Blockchain bridges, the following methodology with 4 step assessment is followed. The sequence of steps given in Table 3 is used to identify the vulnerabilities, starting from identifying existing solutions and understanding the purpose of the solution to exploring the risk involved and proposing possible mitigations for each identified issue. The final step is to identify the proposed solution's limitations and areas where it might not be workable.

Table 3. Overview of six-step assessment methodology

Step	Process
1	➢ Define the components of bridge architecture considered
2	➢ Define the purpose of the bridge solution ➢ Define the goal of the solution ➢ Identify the target system and use case for this solution
3	➢ Identify the components in this solution ➢ Check on components being used in this solution ➢ Categories the solution
4	➢ Define the risks involved in this solution ➢ Categorize the type of risk ➢ Summarize the implications
5	➢ Define possible mitigations for each of the solutions ➢ Define possible risk aversion methods ➢ Identify the limitations
6	➢ Review the system again ➢ Investigate if the mitigations further have any risks involved

4. Result and Findings

Blockchain bridge solutions are of a broad spectrum. Each solution has its purpose and where it can be implemented. Moreover, each solution need not be an independent solution. There can be solutions coupled together to achieve the means of interoperability. For example, a notary scheme can complement an HTLC (Hash Time Lock Contracts), where a notary scheme would help by acting as an intermediary, and an HTLC can help ensure the atomicity of the cross-chain asset transfer. Thus, these solutions should be complementary to one another depending on the design of the interoperable solution and the chains involved and the degree of decentralization targeted. In this study, risks

regarding multiple solutions are discussed. A typical threat model consists of the following steps: (i) List of Assets, (ii) Point of Entries, (iii) Attackers Model, (iv) List of Threats, and (v) Mitigation Plan, as shown in Fig. 3.

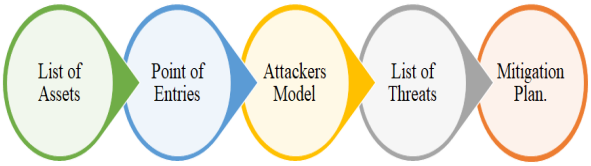

Fig. 3: Steps of the typical threat model

4.1 Asset List

Asset for any program or system can be considered as the primary target. An Asset is a valuable entity in the system which attackers would want to modify or get access to benefit their intents. Table 4 Shows the list of assets identified.

4.2 Points of Entry

Entry points are target areas to begin exploits. Points can vary right from smart contract vulnerabilities to private key access through social engineering. Below are the entry points for the bridge system considered:

Contract Interaction: Contract interaction is the most common way to enter and exploit contract vulnerabilities. One of the biggest attacks, the Dao attack[40] is through smart contract interaction using its vulnerabilities.

Key Generation and Storage: Private key leaks directly relate to controlling or upgrading the upgradable contract with is_admin options. Wallet generation is also crucial, as vulnerable key generation libraries might provide exploiters with privileged functions that could leak private keys. According to certik, a Web3's leading smart contract auditor, during the year 2022, around $273.9 million were lost due to compromised private keys (Crypto Wallet Security Incidents, 2023) .

Relayers: The relay server's primary aim is to relay information from the source to the destination chain, which can then further be used for the verification of data. This can be facilitated using individual relayers or a separate trusted program that does this. This could be an entrance point for threat types, including DoS (Denial of Service).

Chain vulnerability: A bridge solution usually will have source and destination chains. Besides the vulnerabilities in the bridge, vulnerabilities of the chain itself can cause a loss of funds.

Table 4. List of Assets

Asset Name	**Description**
Bridge Contract	Smart contract deployed on one or all the chains to facilitate the transfer/lock of assets.
Relayer application/Client	An application or client that relays information/blocks from one chain to another
Relay Contract	Smart contract on the chain that manages relayed information and verifies information when provided
Chains involved in interoperability.	Actual source and destination blockchains are involved in the interoperability.
Key files	Cryptographic key pair used for singing. It can be of users, relayers or the admin of the contract.
Client application	Can be a user interface or wallet management application that helps in interacting with the chain or contract.

4.3 Threat Agents

Threat agents define the potential attackers and their motives, including capabilities. It is crucial to identify users of a system and their roles and capabilities because in threat modelling, it is considered a threat that originated from either internal users or external agents. Based on that distinction, there can be two types of agents.

Users: End-users of bridge applications which interact with the contract or chain to transfer assets.

Contract Administrators: Contracts, as discussed, can be upgradable and could have some methods that require is_admin privilege. These super-users will have access to crucial functionalities like upgrading the contract, managing or withdrawing funds from the contract, or even being the owner of a custodian wallet.

Motives: The majority of attacks on bridges are for financial gain. Only a few attackers return the funds as their motive would be to find and test the vulnerabilities as a security researcher.

5. Risk and Mitigations

This study explores the risks from the interoperability solution point of view. Each solution, like Relays, HTLCs, Notary Schemes, and Smart Contracts, is explored. Below are the observed risks involved in the solutions, and Table 5 represents the same type of solution along with the issue and adverse consequence(s).

Table 5. Blockchain Bridge modules with associated risks and mitigations

Framework	Threat / Risk	Mitigation
Notary Scheme	Centralization Single point of failure Trust in single third party	Decentralized notary schemes Federated Two way pegs SPV proofs Using HSM (Hardware security module)
Relays	DoS attack Malicious relayers	Multiple Relays Scope for relayers to report and get rewarded
HTLC	Privacy Leaks Wormhole Attacks	Anonymous multi-hop locks Privacy-preserving atomic swaps
Smart Contract	Re-entrancy Permanent Lock of Assets	Use of smart card analytical tools like SLITHER[41] , MYTHX [42] Smart Contract Audit by external agencies
Genetic	Eclipse Attacks Social Engineering Public Key Spoofing	Use standard key storing protocols or use hard wallets for admin privileges.

5.1 Relays

We know that relay's primary work is to pass on data from one Blockchain to one Blockchain to another. This usually can be done either by a dedicated server listening to the host Blockchain, or it can be in the form of the users where users who wish to have cross-chain assets or data transfer can relay over the messages. While there are advantages and disadvantages in both methods like, relay as an application would eliminate the need for users to submit transactions manually, and users as clients would mean probable dependence on transporting messages across,

Moreover, the smart relay contract on the chain would also have to reward the relayers for motivating them to submit blocks. If there is a relay application as a client to submit transactions, a single relay service would lead to several issues such as a single point of failure and thus halt the cross-chain transaction validation (such as verifying a burn transaction on Chain A so that new assets can be unlocked on chain B). A single source of relaying would also increase the chances of a Denial of Service (DoS) attack. This can be averted to a point where multiple relays could be employed. It is also crucial to consider their usage of as not to flood the contract; instead, have a HA (highly available) deployment for the relay system with active and passive relayers.

5.2 Notaries

Notaries are the third-party intermediates that facilitate cross-chain communication, such as asset and data transfer. Notary schemes are more of a trusted witness of a contract among multiple untrusted parties. One major advantage of notaries is that they are not complex to implement and maintain, and not many changes are required in the chains involved either at a protocol or smart contract level. Since notaries act as mediators of transactions between multiple chains, it can become a central point of failure in case of internal adversaries like crashes, downtime or misuse of funds and external threats like theft of funds or exploiting the deployments. A group of notaries can be involved and have a consensus amongst themselves.

5.3 Hash Time-Locked Contracts

HTLCs (Hash Time-Locked Contracts) are the approach usually used during cross-chain atomic asset transfer. It is more of an agreement to produce a payment on a conditional basis by providing certain cryptographic proof to ensure atomicity. It is achieved with a combination of time locks and hash locks. HTLCs are vulnerable to multiple risks like DoS, a kind of attack making the service inaccessible to required users. In the case of HTLC, there could be a case where the swap could not be performed as the one who initiated the method could be in control of abortion too. HTLCs are vulnerable to wormhole attacks where two malicious nodes are involved, and one tunnels data packets to another side; this could disrupt further communication between the parties involved in the atomic swap. HTLCs also suffer privacy leaks, such as the path of the payment meaning the identities involved are visible[43].

5.4 Bridge Smart Contracts

Bridge smart contracts are programs run by the network

that facilitates cross-chain communication. A vulnerability in a smart contract can be of many forms, such as a reentrancy vulnerability. A smart contract defines how the state change happens. The state changes after the successful execution of the contract. There can be a case where the attacker can use the intermediate state in between the state change and can have multiple repeated calls. Another such vulnerability is with locking the tokens in the contract. As discussed previously, bridge contracts can be used to burn or lock assets on one side and then unlock them on the other side. The tokens locked can be later used to redeem the lock or burn the tokens that were previously bridged. In this case, there is a risk of getting funds locked forever, especially when the unlock is done via another contract. This happens when the contract for unlocking or withdrawal is destroyed, EVM does a SELFDESTRUCT, and the bridge contract cannot use the funds further.

5.5 Custody of assets

Blockchains are usually isolated and are designed in such a way that there is no cross-chain communication facilitated. Thus comes third-party intermediaries who take the responsibility of transferring assets across. But this is against the philosophy of decentralization because the third party can act as a bottleneck trust. One such crucial aspect is the holding of assets. As mentioned previously, this third party can be a combination of a chain wallet and third-party servers checking on state changes or it can be a mix of smart contracts on the chain and a relay system transferring blocks across while transferring assets from one chain to another. While using smart contract solutions seems obvious, it should be noticed that certain chains like Bitcoin[14]. In such cases, bridge implementations usually have the custodian hold or manage all the assets. This poses unconditional access and power over how those assets can be handled, thus providing a chance for theft or freezing of the funds. Thus, the trust is on the custodian to lock and unlock the assets. The possible way to mitigate this issue is to have a committee of custodians and a multi-signature approach where the majority of committee members have to sign the transaction to perform actions on funds being bridged.

6. Case Study

This study takes a case study evaluation approach to evaluate the analysis and proposals. The case study analysis approach is considered as this study focuses not just on a single risk and proposal of mitigation to that solution but a series of components and flows. A use case is considered with a simple cross-chain asset transfer and is analyzed for probable risks on the path and how the proposed solutions help.

Test Premise

It is assumed that the Blockchains that are involved in the discussion are sound enough in the lines of security with proper consensus mechanisms in place. For the purpose of the case study, a quotidian use case is considered to analyze security risks. The Block diagram in Fig. 4 represents the participants involved, including chain A (C_A), chain B (C_B), Participant A of chain A (Pa), Participant a of chain B(Pb) and a relayer R and a Client C and smart contract (SC). A cross-chain transfer, in this case, can be defined as a transaction that gets carried out on both ChainA and Chain B in order to have assets or data transferred across, as shown in Eq. 1.

$$CrossChainTransfer : Chain_A \rightarrow Chain_B \quad (1)$$

6.1 Experimental Results

Case Flow

The types of flows involved can be categorized into three. Relay flow(Fetch Block, Deliver Block to SC), User initiated flow(Construct Proof, Trigger cross-chain asset transfer), transaction flow (transfer asset, lock asset, unlock or burn asset). We consider a token of amount X being transferred from chain A to chain B with relayer R relaying messages from chain A to Chain B's contract SC which would be used to verify cross-chain requests. While it is possible that information can flow in both ways, we only consider a cross-chain transfer from chain A to chain B.

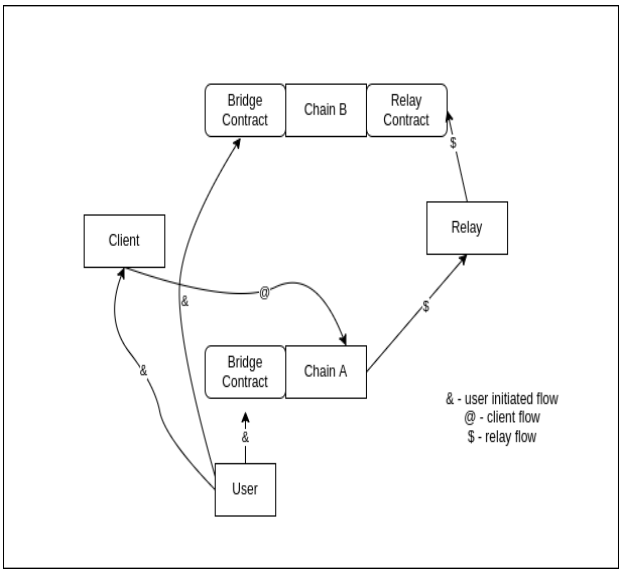

Fig. 4: Participants and flows in cross-chain txn

Case Logic Implementation

To design this use case, a basic cross-chain asset transaction model is considered. Every chain transaction that facilitates will have the following parameters. Type of transaction, Amount, Embedded Proof of Lock. Every Relay message will be of format chainId, blockNumber, and transaction. The word "message" here is used for the

relay as it doesn't perform any cross-chain transfer by itself but facilitates cross-chain transactions like validating the proof submitted by the client. To assess the security risks, a threat model is considered so that for a given flow and topology, we can define the risks involved and check if the countermeasures provided help in mitigating them[44] and enhance the sustainable project in blockchain[45]. Fig. 5 represents the flow of the threat model considered. Below are the identified risks and how the proposed solutions will help mitigate them. If we consider the sequence of flow. There would initially be a call to a smart contract to transfer an asset, further validating data and locking the token as represented by Eq. 2.

$$txn_{transfer} \rightarrow validate() \,\&\, lockToken() \quad (2)$$

Locked Ether

As seen above, the transaction (transfer) internally does a lock of assets, which moves assets from the user to the chain account. The reason behind locking the asset is to make the funds un-usable in chain A and mint a similar amount in chain B and later unlock it further when required during a reverse cross-chain asset transfer case using a *unlock* or *withdraw* method. Smart contracts in Ethereum also have accounts and have the capability to hold funds. In our case transfer asset function, which receives the funds, would be a payable method, indicating it can take in funds to the contract's account. But there can be various cases where the funds cannot be unlocked. One of them is the case where the bridge contract depends on another contract which was destroyed using the SELFDESTRUCT. If the bridge contracts withdrawal needs an external contract that was destroyed, funds cannot be further taken, thus locking the assets in the chain as shown in Eq. 3 and 4.

$$txn_{mint} \rightarrow verifyProof() \,\&\, MintAsset() \quad (3)$$
$$txn_{trasnferBack} \rightarrow unlockAsset() \quad (4)$$

Malicious Relaying

As the smart contract in chain A locks the tokens, it's the relayer that would relay the messages further to the relay contract in chain B for validation of the contract. The relay data is used against the data provided by the client as proof of asset lock or burn in the source chain. However, there is a case where the relay R would transmit blocks that are either not confirmed or maliciously formed blocks. Even though the relay contract would do basic validation, it will be limited to checking if the current block provided has a hash of the existing last block in the contract and other cryptographic integrity validations but not the data itself. Attackers can then call the *mint* method on Chain B with counterfeit proof of assets being burnt on Chain A, which would be valid when validated by the bridge contract against the relay data. A solution thus would be a case where multiple such relayers could be used to transmit data from chain A to relay contract on chain B. Furthermore, a mechanism where relayers validate the provided data and incentivize them for such and punish the malicious relay would help mitigate this issue as given in Eq. 5.

$$C_a(msg, chainId, block) \rightarrow C_b.relayContract.save() \quad (5)$$

Single Point of trust

It was defined that both the contracts in the chain have smart contract abilities and, thus, can lock and unlock tokens on either side. However, not every Blockchain has smart contract capability. In Fact, the infamous Bitcoin[14] does not have smart contract capabilities. To analyze that scenario, we consider chain A not having SC capabilities to lock the assets. Traditionally that is the case where custodian schemes come into play. In this case, custodian accounts should be manually trusted, unlike in the case of smart contracts where code facilitates trust. Thus the custodian becomes the single point of trust and failure. A malicious custodian having sole control over the custodian account can steal the funds. To avert this issue, there can be a group of custodians with a multi-sig wallet, as mentioned in Section 4, and all of them signed to have funds transferred. Even in a case where we consider the custodian non-malicious, it would still be prone to loss of private key or theft. Using an HSM module as shown in Fig. 5, would help access the keys securely when required.

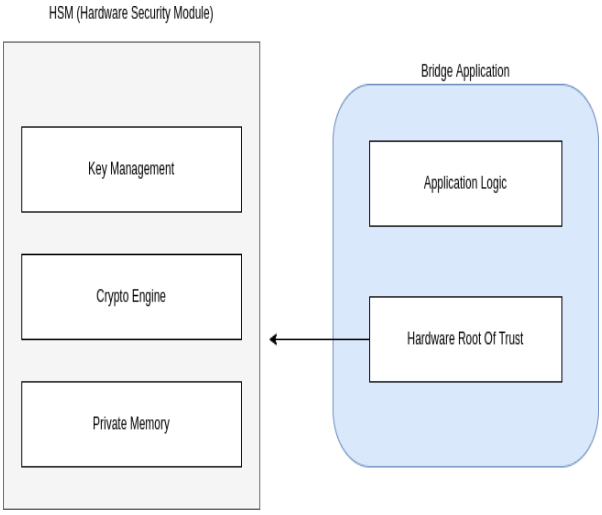

Fig. 5: Use of HSM in bridge applications

7. Conclusion and Future Recommendations

This research paper has been comprehensively analysed interoperable frameworks for Blockchain bridges and their functionality. By exploring the utilization of Relays, HTLCs, Notary Schemes, and Smart Contracts within different solutions, we have identified the associated risks and vulnerabilities while also presenting effective mitigations. To enhance the analysis of security risks inherent in Blockchain bridges, a comprehensive threat

model and case study analysis were proposed. This research aims to equip Blockchain bridge developers with an understanding of the risks involved when employing common frameworks, enabling them to make informed decisions regarding framework combinations based on their specific use cases. Additionally, the provided mitigations serve as practical measures for developers to mitigate these risks. Moreover, this study assists users in comprehending the solutions offered when initiating cross-chain transactions.

Moving forward, it is essential to consider the comprehensive exploration of risks associated with interoperable frameworks, considering their combined deployment rather than examining them in isolation. Standardization of interoperable frameworks is a pressing need in the field. Establishing a common cross-chain communication model, defining recommended cryptographic protocols, or developing a unified framework template are potential avenues for future research and industry collaboration. These efforts would contribute to developing a robust and secure ecosystem for Blockchain bridges. While this study went with a component-based threat analysis along with a case study, different threat modellings like STRIDE or DREAD, or Attack Trees can be used. This study lays the groundwork for future research, industry collaborations, and the adoption of best practices to ensure the secure and efficient functioning of cross-chain transactions.

Reference

1) K.W. Prewett, G.L. Prescott, and K. Phillips, "Blockchain adoption is inevitable—Barriers and risks remain," *Journal of Corporate accounting & finance*, **31**(*2*), 21-28, (2020). https://doi.org/10.1002/jcaf.22415.
2) L. E. Whitman, D. Santanu, and H. Panetto, "An enterprise model of interoperability," *IFAC Proceedings,* **39**(*3*), 609-614, (2006). https://doi.org/10.3182/20060517-3-FR-2903.00311.
3) B. Pillai, K. Biswas, Z. Hóu, and V. Muthukkumarasamy, "Burn-to-claim: An asset transfer protocol for blockchain interoperability," *Computer Networks*, **200**, 108495, (2021). https://doi.org/10.1016/j.comnet.2021.108495.
4) V. Buterin, "Chain interoperability," *R3 Research Paper*, **9**, (2016).
5) P. Lafourcade, and M. Lombard-Platet, "About blockchain interoperability," *Information Processing Letters*, **161**, 105976, (2020).
6) B. Pillai, K. Biswas, and V. Muthukkumarasamy, "Cross-chain interoperability among blockchain-based systems using transactions," *The Knowledge Engineering Review*, **35**, e23,(2020). DOI: https://doi.org/10.1017/S0269888920000314.
7) L.W. Cong, G.A. Karolyi, K. Tang, and W. Zhao, "Value premium, network adoption, and factor pricing of crypto assets," *Network Adoption, and Factor Pricing of Crypto Assets*, (2021).
8) R. Belchior, A. Vasconcelos, S. Guerreiro, and M. Correia, "A survey on blockchain interoperability: Past, present, and future trends," *ACM Computing Surveys (CSUR)*, **54**(*8*), 1-41, (2021). https://doi.org/10.1145/3471140.
9) Extropy.IO, "Solana's Wormhole Hack Post-Mortem Analysis," n.d. https://extropy-io.medium.com/solanas-wormhole-hack-post-mortem-analysis-3b68b9e8 8e13 (accessed Jun 26, 2022).
10) R.B. Sigalos, and MacKenzie, "Hackers have stolen $1.4 billion this year using crypto bridges. Here's why it's happening," CNBC, 2022. Available at: https://www.cnbc.com/2022/08/10/hackers-have-stolen-1point4-billion-this-year-using-crypto-bridges.html (accessed August 31, 2022).
11) T. Hardjono, "Blockchain gateways, bridges and delegated hash-locks," arXiv preprint arXiv:2102.03933, (2021).
12) M. Borkowski, D. McDonald, C. Ritzer, and S. Schulte, "Towards atomic cross-chain token transfers: State of the art and open questions within tast," Distributed Systems Group TU Wien (Technische Universit at Wien), Report, 8, (2018). DOI: 10.13140/RG.2.2.10769.48489
13) D. Singh, and A Singh, "Role of Building Automation Technology in Creating a Smart and Sustainable Built Environment." *Evergreen*, **10**(*1*), 412-420, (2023). https://doi.org/10.5109/6781101.
14) S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Decentralized business review*, 21260, (2008).
15) G. Wang, and M. Nixon, "Sok: Tokenization on blockchain*," In Proceedings of the 14th IEEE/ACM International Conference on Utility and Cloud Computing Companion*, 1-9, (2021). DOI: 10.1145/3492323.3495577.
16) Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," *In 2017 IEEE international congress on big data (BigData congress)*, 557-564, IEEE, (2017). DOI: 10.1109/BigDataCongress.2017.85.
17) A. Verma, P. Singh, and N. Singh, "Study of blockchain-based 6G wireless network integration and consensus mechanism," *International Journal of Wireless and Mobile Computing*, **21**(*3*), 255-264, (2021).https://doi.org/10.1504/IJWMC.2021.120906.
18) Vernadat, F. B., "Interoperable enterprise systems: architectures and methods," *IFAC Proceedings,* **39**(*3*), 13-20, (2006). https://doi.org/10.3182/20060517-3-FR-2903.00010.
19) S. Khan, M.B. Amin, A.T. Azar, and S. Aslam, "Towards interoperable blockchains: A survey on

the role of smart contracts in blockchain interoperability", IEEE Access, 9, 116672-116691,(2021).DOI: 10.1109/ACCESS.2021.3106384.

20) A. Hope-Bailie, and S. Thomas, "Interledger: Creating a standard for payments," *In Proceedings of the 25th international conference companion on world wide web*, 281-282, (2016). https://doi.org/10.1145/2872518.2889307.
21) H.T. Vo, Z. Wang, D. Karunamoorthy, J. Wagner, E. Abebe, and M. Mohania, "Internet of blockchains: Techniques and challenges ahead, " *In 2018 IEEE international conference on internet of things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData)*, 1574-1581, IEEE, (2018). 10.1109/Cybermatics_2018.2018.00264
22) I.A. Qasse, M. Abu Talib, and Q. Nasir, "Inter blockchain communication: A survey," *In Proceedings of the Arab WIC 6th Annual International Conference Research Track* , 1-6, 2019. https://doi.org/10.1145/3333165.3333167.
23) R. Han, H. Lin, and J. Yu, "On the optionality and fairness of atomic swaps," *In Proceedings of the 1st ACM Conference on Advances in Financial Technologies*, 62-75, (2019). https://doi.org/10.1145/3318041.3355460.
24) S. Johnson, P. Robinson, and J. Brainard,"Sidechains and interoperability," arXiv preprint arXiv:1903.04077, (2019).
25) T. Koens and E. Poll, "Assessing interoperability solutions for distributed ledgers," *Pervasive and Mobile Computing*, *59*, 101079, (2019). https://doi.org/10.1016/j.pmcj.2019.101079.
26) A. Zamyatin, M. Al-Bassam, D. Zindros, E. Kokoris-Kogias, P. Moreno-Sanchez, A. Kiayias, and W.J. Knottenbelt, "Sok: Communication across distributed ledgers," In Financial Cryptography and Data Security: 25th International Conference, FC 2021, Virtual Event, March 1–5, 2021, Revised Selected Papers, Part II 25 (pp. 3-36). Springer Berlin Heidelberg, (2021).
27) A. Singh, K. Click, R.M. Parizi, Q. Zhang, A. Dehghantanha, and K.K.R. Choo, "Sidechain technologies in blockchain networks: An examination and state-of-the-art review," *Journal of Network and Computer Applications*, 149, 102471, (2020). https://doi.org/10.1016/j.jnca.2019.102471.
28) N. Kannengießer, M. Pfister, M. Greulich, S. Lins, and A. Sunyaev, "Bridges between islands: Cross-chain technology for distributed ledger technology", (2020). https://hdl.handle.net/10125/64394.
29) Monika, and R. Bhatia, "Interoperability solutions for blockchain," *In 2020 international conference on smart technologies in computing, electrical and* electronics (ICSTCEE),381-385. IEEE, (2020). DOI: 10.1109/ICSTCEE49637.2020.9277054.
30) B. Dai, S. Jiang, M. Zhu, M. Lu, D. Li, and C. Li, "Research and implementation of cross-chain transaction model based on improved hash-locking," In Blockchain and Trustworthy Systems: Second International Conference, BlockSys 2020, Dali, China, August 6–7, 2020, Revised Selected Papers 2 (pp. 218-230). Springer Singapore, (2020). https://doi.org/10.1007/978-981-15-9213-3_17.
31) S. Ghaemi, S. Rouhani, R. Belchior, R.S. Cruz, H. Khazaei, and P. Musilek, "A pub-sub architecture to promote blockchain interoperability," arXiv preprint arXiv:2101.12331, (2021).
32) R. Lan, G. Upadhyaya, S. Tse, and M. Zamani, "Horizon: A gas-efficient, trustless bridge for cross-chain transactions," arXiv preprint arXiv:2101.06000, (2021).
33) M. Sober, G. Scaffino, C. Spanring, and S. Schulte, "A voting-based blockchain interoperability oracle," *In 2021 IEEE International Conference on Blockchain (Blockchain)* 160-169, (2021). 10.1109/Blockchain53845.2021.00030.
34) H. Su, B. Guo, J.Y. Lu, and X. Suo, "Cross-chain exchange by transaction dependence with conditional transaction method," *Soft Computing*, 1-16, (2022). https://doi.org/10.1007/s00500-021-06577-5.
35) G. Caldarelli, "Wrapping trust for interoperability: A preliminary study of wrapped tokens," *information*, **13**(1),6,(2021). https://doi.org/10.3390/info13010006.
36) M. Madine, K. Salah, R. Jayaraman, Y. Al-Hammadi, J. Arshad, and I. Yaqoob, "appxchain: Application-level interoperability for blockchain networks," IEEE Access, 9, 87777-87791, (2021). DOI: 10.1109/ACCESS.2021.3089603.
37) H. Su, "Cross-chain interaction model in a fully verified way," arXiv preprint arXiv:2106.05463, (2021).
38) T. Haugum, B. Hoff, M. Alsadi, & J. Li, "Security and Privacy Challenges in Blockchain Interoperability-A Multivocal Literature Review," In *Proceedings of the International Conference on Evaluation and Assessment in Software Engineering* 347-356, (2022, June). https://doi.org/10.1145/3530019.3531345.
39) H. Uddin, and MK. Barai, "Will Digital Revolution be Disruptive for the Inclusive Finance in Bangladesh? The Case of the Microfinance Industry." *Evergreen*, **9**(*4*), 909-923, (2022). DOI: 10.5109/6622878
40) M.I. Mehar, C.L. Shier, A. Giambattista, E. Gong, Fletcher, R. Sanayhie, H.M Kim,and M. Laskowski, "Understanding a revolutionary and flawed grand experiment in blockchain: the DAO attack," *Journal of Cases on Information Technology (JCIT)*, **21**(*1*), 19-32, (2019). DOI: 10.4018/JCIT.2019010102.

41) J. Feist, G. Grieco, and A. Groce "Slither: a static analysis framework for smart contracts" *In 2019 IEEE/ACM 2nd International Workshop on Emerging Trends in Software Engineering for Blockchain (WETSEB)*, 8-15, IEEE, (2019). https://doi.org/10.1109/WETSEB.2019.00008.
42) M. Mossberg, F. Manzano, E. Hennenfent, A. Groce, G. Grieco, J. Feist, E. Hennenfent, A. Groce, G. Grieco, J. Feist, T. Brunson, and A. Dinaburg "Manticore: A user-friendly symbolic execution framework for binaries and smart contracts," *In 2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 1186-1189. IEEE, (2019). DOI: 10.1109/ASE.2019.00133.
43) A. Deshpande, and M. Herlihy, "Privacy-preserving cross-chain atomic swaps," *Financial Cryptography and Data Security. FC 2020*. Lecture Notes in Computer Science, 12063, 540-549, Springer, Cham, (2020). https://doi.org/10.1007/978-3-030-54455-3_38.
44) G. Shemov, B. Garcia de Soto, and H. Alkhzaimi, "Blockchain applied to the construction supply chain: A case study with threat model", *Frontiers of Engineering Management*, *7*, 564-577, (2020). DOI: 10.1007/s42524-020-0129-x.
45) A. K. Singh, V. P. Kumar, G. Dehdasht, G., S. R. Mohandes, P. Manu, P., & F. Rahimian, "Investigating the barriers to the adoption of blockchain technology in sustainable construction projects." *Journal of Cleaner Production*, 403 (2023): 136840. https://doi.org/10.1016/j.jclepro.2023.136840