Stroom: an Economically Secured Yield-Bearing BTC

Viacheslav Zhygulin s@stroom.network, Rostyslav Shvets r@stroom.network
v.1.0

Abstract

Stroom is a permissionless, cryptoeconomically secured yield-bearing BTC protocol. stroomBTC is a BTC LST operating on major DeFi chains, including Ethereum, Solana, Base, and key L2s. Stroom utilizes low-risk BTC yield sources (Lightning Network [28], CoreDAO [16] staking, Babylon [1] restaking, BitVM2 [10] Operator staking, and liquidity bridge for the BitVM2) for providing stroomBTC holders with a BTC yield and paying for the protocol's cryptoeconomic [17] security. The slashable stake is supplied to the Stroom validators via restaking [36]: when the validators act maliciously, they face substantial economic penalties. The protocol's design is based on dMPC for Schnorr signatures called FROST [25] to ensure validator accountability, even when BTCs are deposited in yield-generating strategies. Stroom DAO manages the validator set and rules, yield generation and distribution, slashing, and overall protocol governance.

Introduction

Bitcoin's limited scripting capabilities have been the basis of its design, ensuring flawless Bitcoin blockchain security [6]. Simple scripts narrow down a potential attack surface to the point where it is impossible to spend the funds secured by a Bitcoin script in a way that was not intended by the script developer. In contrast, unintentional behavior in Ethereum's feature-rich smart contracts [24] led to many notorious hacks. Such slim Bitcoin scripting capacities, on the one hand, played a crucial role in establishing Bitcoin as the most secure blockchain; on the other, it makes it impossible for Bitcoin to implement DeFi [18] natively on its chain. Therefore, Bitcoin users seeking to utilize their BTC in DeFi were forced to use bridges [4] that are "moving" BTCs to DeFi-friendly chains. Consequently, bitcoin bridging became a vast market.

According to Defilama [3], the total USD value of BTC bridged to various other chains was $24.76B as of February 2025. Only two bridges account for approximately 75% ($18.96B) of that TVL: the WBTC [47] bridge to Ethereum ($12.41B) and the Binance Bitcoin bridge to BSC [12] ($6.55B). This data suggests that most adoption of BTC bridging so far happened with products that usually require users to trust some third party that holds native BTC in their custody.

Bitcoin Bridging Concerns

Users of custodian bridges face two significant issues. The first problem is a result of the centralization of such products. Users must trust the related companies to manage the bridge funds properly. In other words, only those companies guarantee that the BTC amount equals the total supply of wrapped Bitcoin or that wrapped BTC is always redeemable for the underlying native BTC.

The second problem is the bridge's opportunity cost. While creating a wrapped token on the DeFi-enabled chain, native BTC liquidity is stored idly in the Bitcoin blockchain. Opportunity cost comes from the fact that native BTC liquidity stored in the bridge could be used to produce low-risk returns, which in turn can be distributed to the bridge's users. However, most bridges require users to pay extra fees for usage instead of rewarding their customers for supplying TVL with some yield.

Unsurprisingly, various new projects arose that tried to solve these two concerns.

Existing Solutions

Emerging Bitcoin bridges can generally be divided into three categories: proof-of-authority federated, collateralized, and optimistically trustless bridges.

Federation-Based Bridges

In these systems, a federation of validating parties collectively oversees the locking and unlocking of BTC on the Bitcoin blockchain and the minting of tokenized BTC on another blockchain. Unlike centralized bridges, these bridges improve their design by distributing control and responsibility across multiple but still known and trusted participants, removing single points of failure and becoming more permissionless. Usually, the federation obeys some practical byzantine fault-tolerant consensus [33], which is guaranteed to function correctly under the assumption that ⅔ + 1 of federation members behave honestly or strictly follow the bridge protocol. Hence, users of such a bridge still need to trust that a quorum of a supermajority of honest federation members always exists.

Nevertheless, several federation-based Bitcoin bridges are tackling the opportunity cost problem.

Lombard

Lombard [32] is a Bitcoin liquid staking protocol with $1.85B TVL that utilizes the BTC liquidity in the Babylon restaking protocol. Babylon is a platform for employing BTC liquidity as a stake to validate various Proof-of-Stake [34] chains, which pay the stakers for the additional security in return. The rewards are usually paid in a token of the validated chain. Hence, the downs and turns of the broader crypto market conditions can drastically influence Lombard's yield performance.

The protocol creates a wrapped interest-bearing BTC token - LBTC - that is functional on the EVM chains. The bridging and staking operations are managed by several yet-to-be-revealed parties via a multisig scheme, effectively making the Lombard a proof-of-authority federation-based bridge.

Solv Protocol

In contrast to Lombard, Solv [39] utilizes its $2.51B liquidity in several sources of yield. Those sources are Babylon restaking, funding rates on Jupiter perpetual DEX [26], Ethena rewards [23], and Bitcoin L2 staking on CoreDAO and Berachain [2]. Some of the sources, like Jupiter, are riskier but produce higher yields.

The wrapped BTC token is available on various EVM chains. Some liquidity is guarded by custodians like Ceffu [13] and Cobo [15]. Therefore, the Solv protocol can be classified as a mixed bridge — partially centralized and partially federated.

Collateralized Bridges

The most significant innovation of cryptocurrencies is that public blockchains are designed to function under the assumption that all validating nodes are rational, not necessarily honest. This property is achieved by establishing some protocol's financial rules. Under those rules, the potential gain of the validating nodes attacking the network is much lower than the economic losses that attacking nodes will suffer, making the attack economically irrational. In Proof-of-Stake chains, those rules are called slashing conditions [45]. When slashing conditions are appropriately designed, it is in the best interest of the validators to follow the protocol honestly to ensure that their stake is not slashed. Therefore, it is a persistent economic equilibrium for actors to follow the protocol, even though those actors can be pseudo-anonymous like Bitcoin miners.

Several bridges have tried implementing slashing conditions using single or multiple assets as collateral. We are unaware of any collateralized bridge that has attempted to solve the opportunity cost problem.

RenBTC

This already-shut down bridge [35] was collateralized by its native REN token. As the REN token had no utility other than staking it for the bridge operator, it had very shallow liquidity. Thin liquidity was insufficient to accommodate slashings appropriately, nor was REN allowed to use it to reimburse users who suffered economic losses because of the operator's malicious actions. Moreover, if information about the hacking of such a bridge became publicly available, the price of its native token would dump heavily, reducing the utility of REN bonds even further.

There is no publicly available information about the RenBTC hacks, but rumors suggest that some users were scammed through censorship attacks without reimbursement. The project was shut down in December 2022 following the collapse of FTX and its affiliated Alameda Research.

tBTC

Threshold Network [44], which stands behind the tBTC [43], developed a randomized operator-assigning algorithm for allocating particular BTC deposit storage to a subset of validators bonded by native T token stakes. Hence, firstly, the impact of a single validator is decreased. Secondly, it is much harder to coordinate for a collusion in a randomized setting. Still, such a setup is questioned regarding its security since randomization does not entirely eliminate the possibility of operator collusion.

Trustless Bridges (BitVM2)

Introduced in 2024, BitVM2 proposes a framework for building trust-minimized Bitcoin bridges, significantly increasing its security and establishing confidence in the bridge operators. Within that protocol, BTC liquidity is stored not in some simple custodian or multisig vault but in a complex Bitcoin contract that prohibits operators from performing unauthorized withdrawals from the vault. We refer the reader to the original BitVM2 paper [9] for more technical details of its implementation.

The BitVM2 contract is quite a complex Bitcoin script. Integration of that with some Bitcoin yield-generation pool is not yet possible. Therefore, BitVM2 bridges fundamentally cannot solve the opportunity cost problem of bridges. Moreover, to create a substantial incentive for BitVM2 operators to perform redemptions, a significant withdrawal fee must be embedded in the bridge design, making using the BitVM2 bridge more expensive. Running the BitVM2 operator and challenger is a good native BTC yield strategy, but we will explore that in more detail in the corresponding section.

Bitlayer

Initially launched as Bitcoin L2 [8], Bitlayer recently announced the liftoff of a BitVM2-based trustless bridge to the Ethereum blockchain [7]. As of February 2025, this technology is available only on the testnet. There will be no native yield for users of this bridge.

BitVM2 bridges are excellent choices for users seeking maximum security while using their BTCs in DeFi and paying higher costs for enhanced security. For users willing to take slightly higher risks and be compensated for bridging opportunity costs, we present the Stroom protocol.

Stroom Protocol

Stroom aims to advance Bitcoin bridges by solving the opportunity cost problem with enhanced safety. It leverages restaking protocols like Symbiotic [41] and Eigenlayer [19] to achieve greater security and attract deep liquidity collateralization. The sustainable yield on bridged value will be generated by providing BTC native liquidity to low-risk infrastructure protocols like Lightning Network, BitVM2, Babylon, CoreDAO, Stacks [40], and other Bitcoin L2s.

The protocol consists of four parts: interest-bearing stroomBTC token on DeFi chains starting with Ethereum, federated bridge infrastructure, yield-generating strategies, and an economic security layer provided by restaking.

StroomBTC

Stroom's wrapped BTC token is an interest-bearing rebasing token bridged to a DeFi-enabled chain. To achieve integration with other DeFi protocols, Stroom utilizes a dual token model. StroomBTC is a rebasing token with users' balances increasing due to payouts when bridge operators distribute rewards pegged to Bitcoin by a redemption mechanism, allowing users to convert their stroomBTC to native BTC always at a one-to-one ratio, subject to a Bitcoin on-chain fee. The second token, wstroomBTC (short for wrapped stroomBTC), is an ERC-20 [21] wrapper for better compatibility with DeFi. Its balance does not change, but the conversion price of wstroomBTC to stroomBTC is increasing. WstroomBTC is intended to be integrated into DeFi protocols.

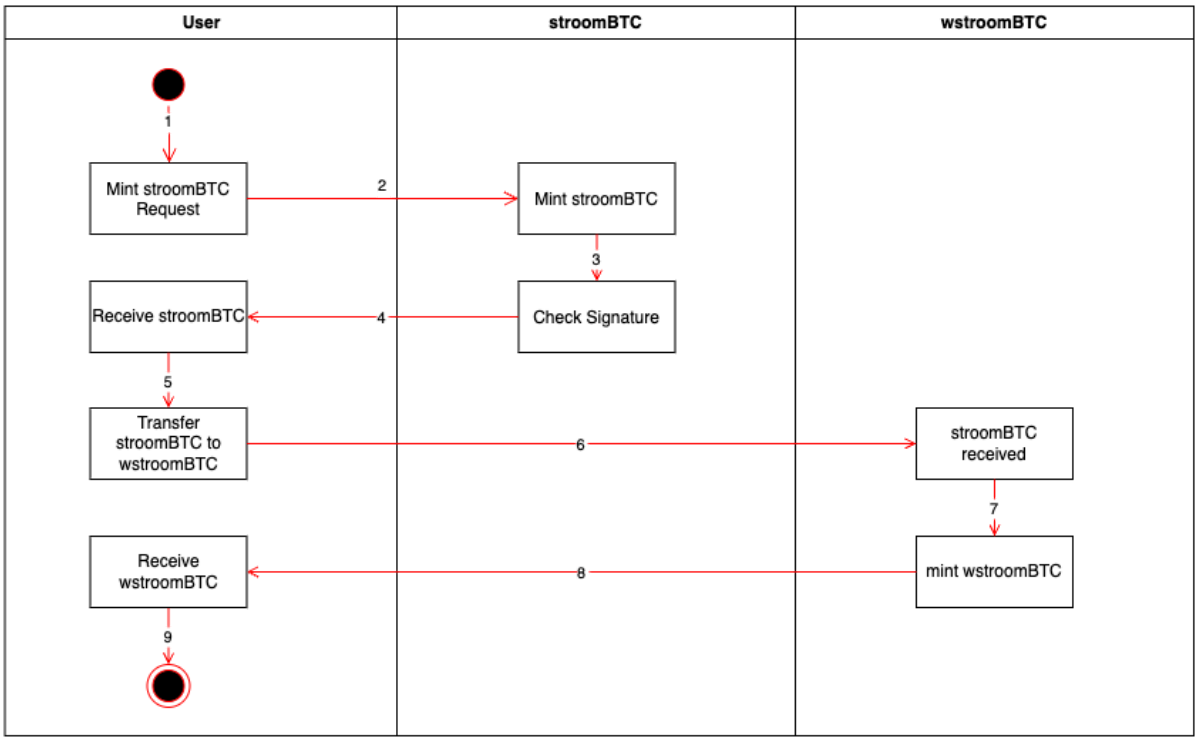

Diagram 1: Minting wstroomBTC

To get wstroomBTC, a user deposits stroomBTC into the wstroomBTC contract in exchange for bwstroomBTC tokens (Diagram 1). This operation can be done simultaneously with the minting of stroomBTC in a single transaction (Diagram 2). Later, the user can unbound the stroomBTC token from the wstroomBTC contract at a rate that is constantly growing due to the accumulation of rewards.

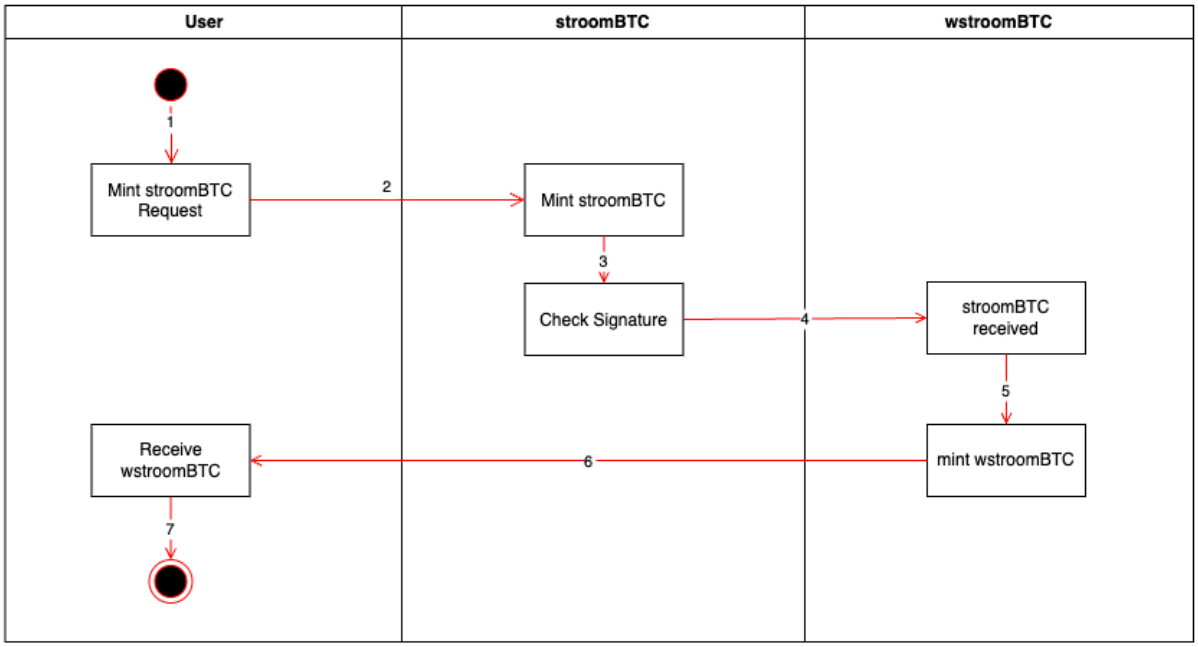

Diagram 2: Minting wstroomBTC Simultaneously with Minting stroomBTC

The yield distribution is performed daily by operators of the Stroom bridge. To do so, operators compute a snapshot of BTC liquidity available in the system after accumulating the yield, compare it with the total amount of stroomBTC, and mint additional stroomBTC as rewards to the stroomBTC contract, which is evenly distributing yields to stroomBTC holders. As a result of this operation, the total supply of stroomBTC matches the total available BTC in the Stroom custody, maintaining stroomBTC's redeemability at a 1-to-1 ratio to the original Bitcoin (diagrams 3 and 4)

The stroomBTC's redemptions back to native BTC are performed permissionless with a low unbounding period, comparable to several Bitcoin block times. The portion of BTC is stored in a multisig wallet and is reserved for processing payouts. In the case of a high surge of redemptions, some BTC liquidity might need to be moved out of Bitcoin yield-generating vaults. Moving funds from the Lightning Network could take a little bit longer, but in the general case, it is just an additional Bitcoin transaction. Hence, the total unbounding period increases from 30 minutes to one hour.

In order to provide capital-efficient safety collateralization of the stroomBTC, Stroom integrates restaking protocols such as Eigenlayer and Symbiotic, starting with Symbiotic. The objective is to attract bridge-securing collateral in significant volumes to increase the safety of the federation-based bridge. We will dive deep into that setup later in the corresponding section.

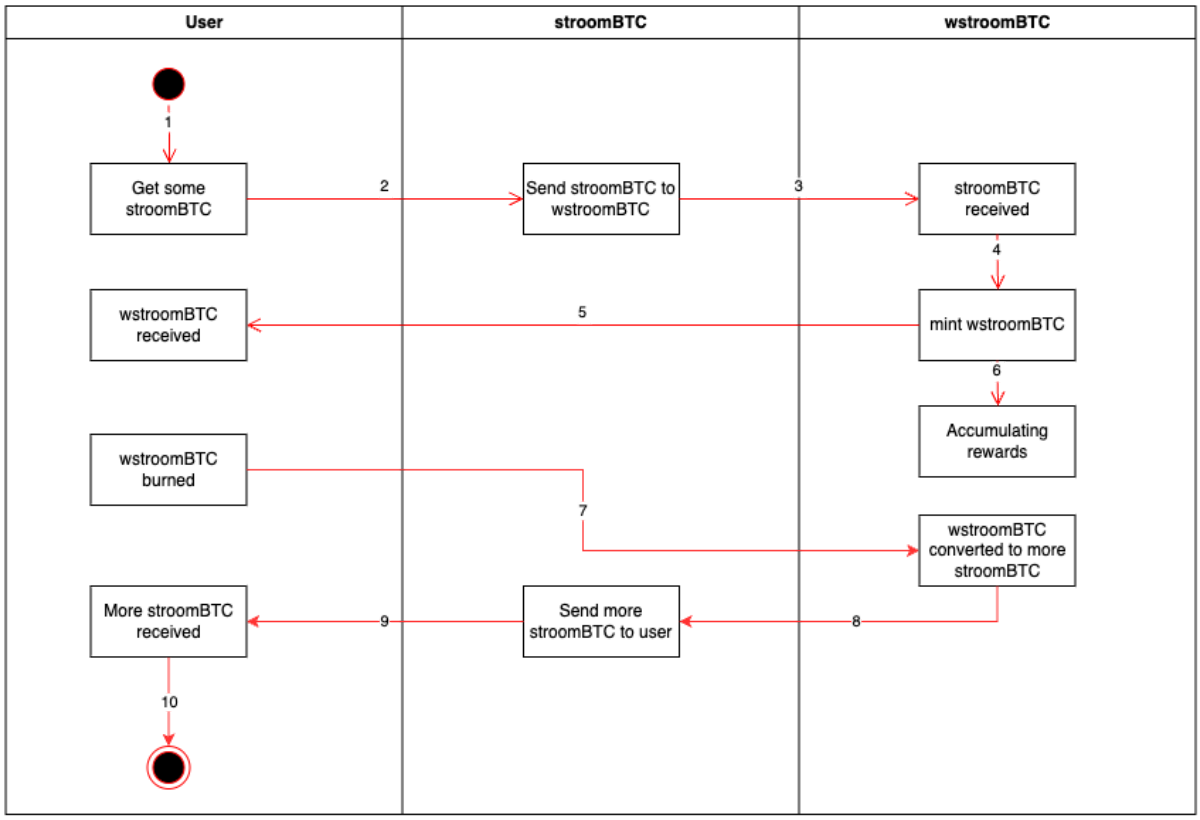

Diagram 3: Depositing to and Withdrawal from wstroomBTC

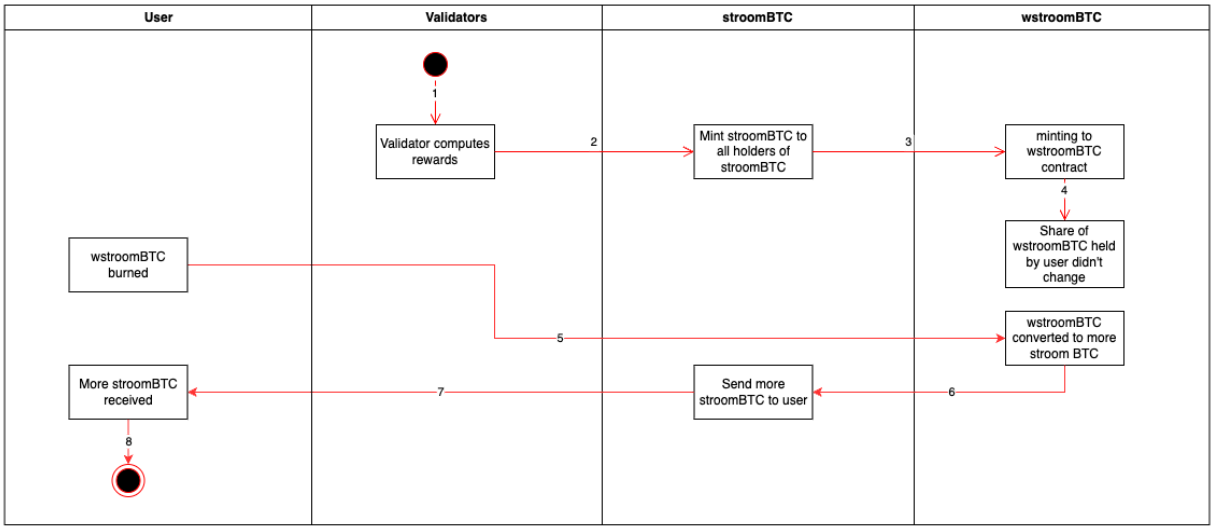

Diagram 4: Posting Rewards to stroomBTC

Stroom Bridge

The bridge is secured by N, which is equal to 10-20 reputable, well-known companies that are professionals in running the validator's business. Each validator, independently from others, checks the validity of the transaction to be executed. For approval and processing, at least K, which is equal to [⅔*N]+1, validating nodes must agree to perform the transaction. The final approving Schnorr signature [38] is collected by running the FROST [25] algorithm by nodes performing several rounds of communication using a closed peer-to-peer Brontide communication protocol [11]. All necessary information for conducting validation is derived from the Ethereum and the Bitcoin blockchains. Hence, there is no need for a global state to be shared privately between validators.

Native BTC deposited to Stroom by users is stored on a federation-controlled Bitcoin multisig wallet. During the wallet initiation procedure, a joint master public key is created. Then, a unique Taproot deposit address [42] is generated for each user, cryptographically derived from the federation's joint master public key and the user's Ethereum address. When spending from that Bitcoin deposit address, the federation signs the transaction with their private keys tweaked by the user's Ethereum address. Taproot scripts check the resulting Schnorr signature against the corresponding user-specific joint public key on the transaction confirmation.

The Ethereum smart contract also executes the minting of stroomBTC using Schnorr signatures. When the stroomBTC contract performs a mint transaction, the corresponding minting details signed by the federation must be supplied into the contract. Those details include the BTC deposit transaction hash, output number, amount, and Ethereum destination address. The contract checks the validity of the Schnorr signature produced by the federation against the federation's joint master public key.

The validator handles cross-chain messaging in case of redemption. They detect the user-initiated stroomBTC burn transaction and must execute the corresponding payout transaction. When performing the mint operation, validators create an approving signature to be supplied into the stroomBTC contract and share it with the user. Yet, it is the user's duty to submit the mint transaction to the contract.

To mint stroomBTC, a user registers his Ethereum address in the stroomBTC contract with a special transaction. Now, bridge operators can derive that transaction from the Ethereum blockchain and use the user's Ethereum address to generate private keys for the unique Taproot deposit address that is cryptographically bound to the user's Ethereum address. Those private keys are stored locally in the validating nodes' encrypted databases. When BTC funds are credited to the user's deposit address, Stroom validators fetch the amount, Bitcoin transaction hash, corresponding output number, and the user's Ethereum address. Collected information is assembled into the minting message and is signed with federation members' master private keys. The federation then shares the message and the signature with the user. After that, the user supplies the message and the signature to the contract, which checks the validity of the signature and mints the corresponding amount of stroomBTC to the user's

Ethereum address if the signature is valid. During the stroomBTC minting, Bitcoin transaction hash and output are marked in the contract as already processed to prevent double mints by the user.

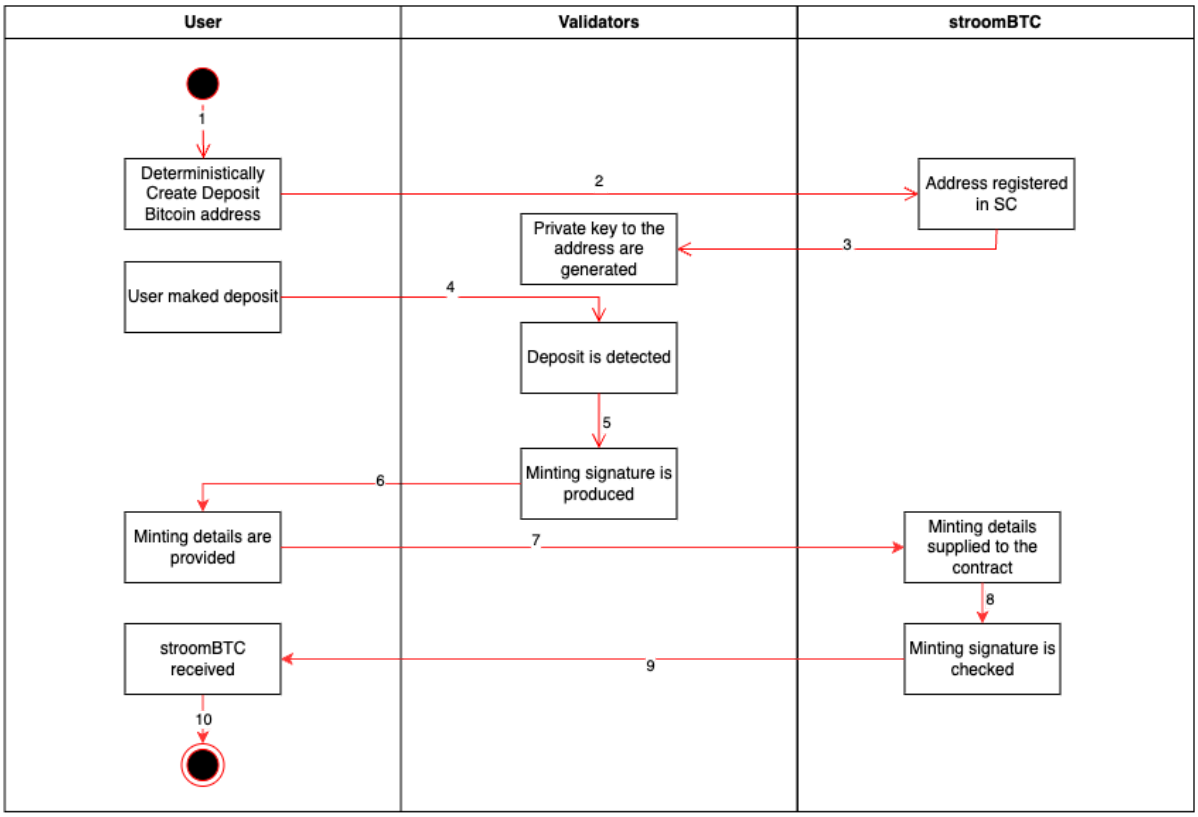

Diagram 5: Depositing BTC to Get stroomBTC

To maintain the stroomBTC peg to BTC, Stroom enables users to redeem their stroomBTC directly to BTC at a one-to-one ratio. For redemption, a user burns stroomBTC on the Ethereum side, specifying the payout destination Bitcoin address in the burn transaction. The validity of the Bitcoin address is checked in the contract to mitigate redemptions to a void destination. Each burn transaction has its unique redemption ID, which sequentially increases with each additional burn transaction. Validators detect the burn transaction and create a payout transaction funded with available UTXOs in the wallet. To prevent double-spending, the redemption ID is written to the sequence number [5] of this payout transaction. The protocol forbids the validators from posting two or more transactions with the same redemption ID into the Bitcoin wallet. Small fees will be deducted from the redemption transaction balance to compensate for related Bitcoin on-chain transaction costs. Any validator posts the resulting transaction to the Bitcoin blockchain.

Low-Risk Sources of BTC Yield

Instead of storing 100% of BTC liquidity in multisig vaults, a portion of that liquidity can be allocated to risk-minimized Bitcoin staking mechanisms. Such an approach allows customers to generate additional yield and compensate for lost opportunity costs related to using the bridge. Several technologies create possibilities for risk-minimized BTC yield.

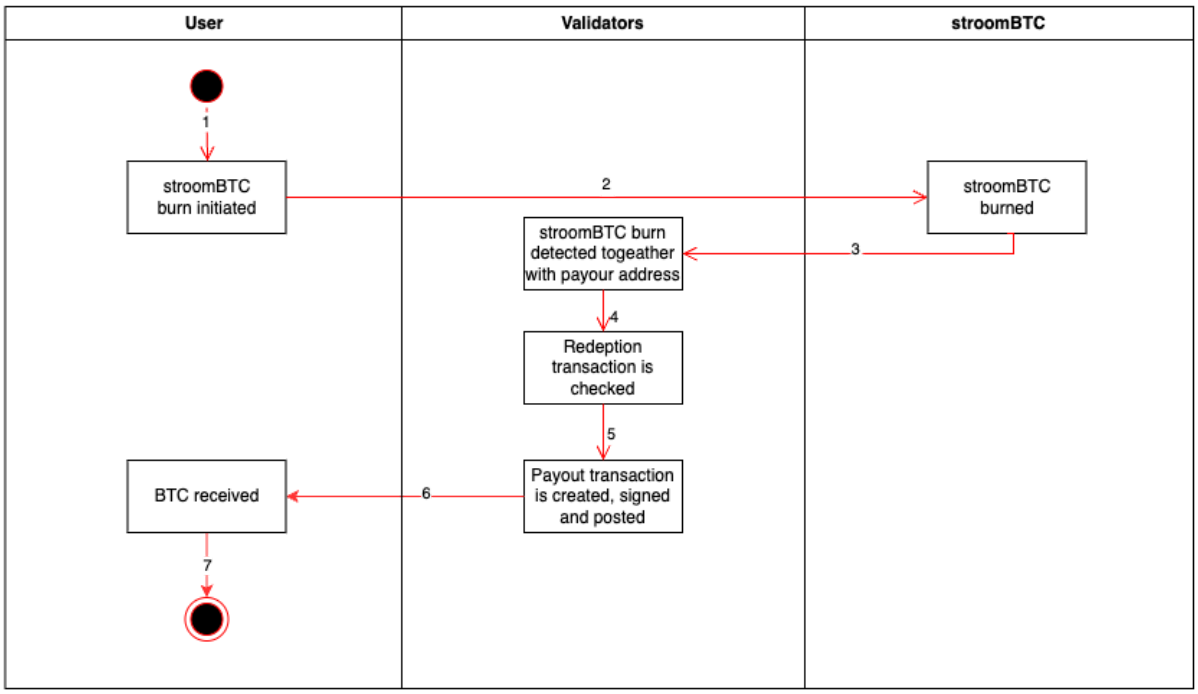

Diagram 6: Redeeming stroomBTC to BTC

Lightning Network

Lightning Network (LN) is a scaling solution to the Bitcoin blockchain explicitly designed for payments. Introduced in 2015, this technology shifts the paradigm of transaction confirmation in distributed networks. At its core, it is a P2P network of nodes that are connected to each other through payment channels. Any node can find a path to any other node in the network and securely route the payment along that path. Since LN directly sends the value, a small fee can be efficiently subtracted from each payment in favor of intermediary nodes who participated in the confirmation of the transaction. This technology opens up an opportunity for a new infrastructural risk-minimized business - routing payments in LN.

In the context of Stroom, it is essential that LN can provide low-risk yield denominated in BTC on liquidity stored in the Stroom bridge. Some initial data suggests [37] that Lightning Network routers can generate up to 6% APR in native Bitcoin yield [29]. As of February 2025, around 5000 BTC were stacked in LN channels. Moreover, the Lightning Network has just added support for USDT [27].

The setup for conducting Lightning yield is as follows. Several Stroom-enabled LN nodes are set up and initially hosted within the custodian's infrastructure. They will later be decentralized using Multisig Lighting by VLS [46]. The governance process decides how much liquidity can be delegated to a particular node. Based on that decision, a node can initiate the deposit process from the Stroom's BTC treasury to the Lightning node. Operators check the validity of the claim and perform the deposit.

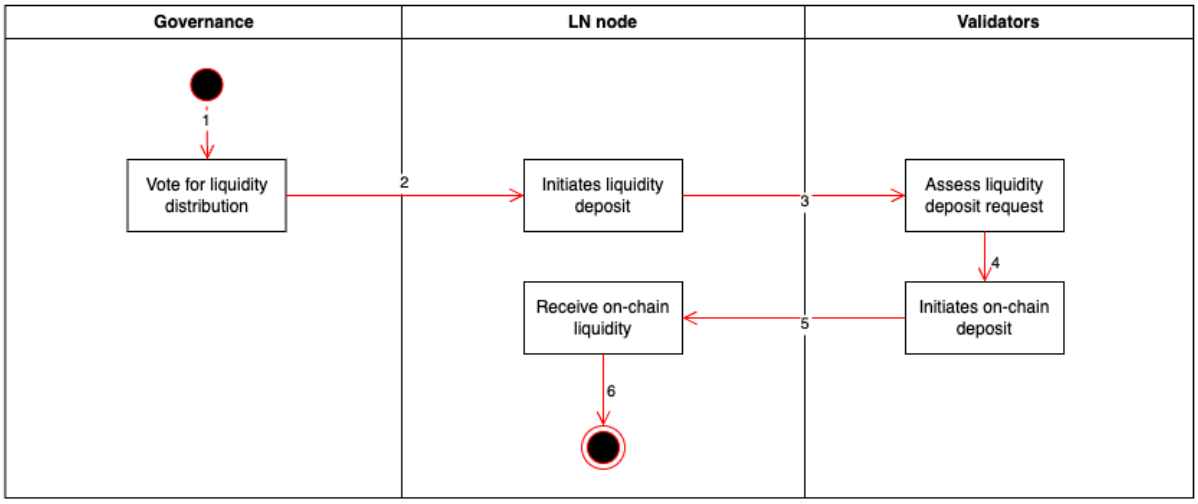

Diagram 7: Deposit to Lightning Node

When a node is funded by BTC, it starts using this liquidity to open channels with other Lightning nodes. Opening enough channels with proper peers guarantees that Lightning payments flow through the node, streaming between channels. The node collects a small fee on each payment, e.g., 0.1%, increasing the total liquidity in the node's channels. Attracting enough inbound liquidity and constantly rebalancing channels is performed by the node operator to improve capital efficiency. In the decentralized setup, all these operations will be checked and approved by Stroom validators. The node operator cannot withdraw funds from the node without the validators' authorization.

Liquidity can stay in the node until governance voters provide enough votes for that node to keep the liquidity. That decision is based on the node's performance, but more about that in the corresponding section related to governance. If there are not enough votes, the node must close some of the channels and return the BTC liquidity back to the multisig wallet. The second possible scenario when the node needs to return liquidity back to the bridge's wallet is when the stabilization pool for processing redemptions falls below the parametrized threshold value, e.g., 5% of all Stroom's BTC liquidity. Channel closure is usually performed by a single on-chain Bitcoin transaction, which can be processed within 10-30 minutes. It is up to the node operator which channels to close, but it is logical to close the worst-performing channels.

While LN nowadays is the most developed and adopted technology for providing native BTC yield, it is not the only one. There are other opportunities to explore.

BitVM2 as a Source of Yield

As stated in the previous section, running BitVM2 operators and challengers can provide a sustainable BTC yield. BitVM2, by its design, needs a proper set of operators who conduct peg-out transactions and challengers who keep track of the operators' transactions and guarantee that operators post no fraudulent transactions. Both operators and challengers require extensive BTC stakes in order to perform their duties, and they are genuinely rewarded for their operations. That creates an opportunity to establish a staking business on top of an operator and challenger that can provide native Bitcoin yield for the BTC staked by stakers.

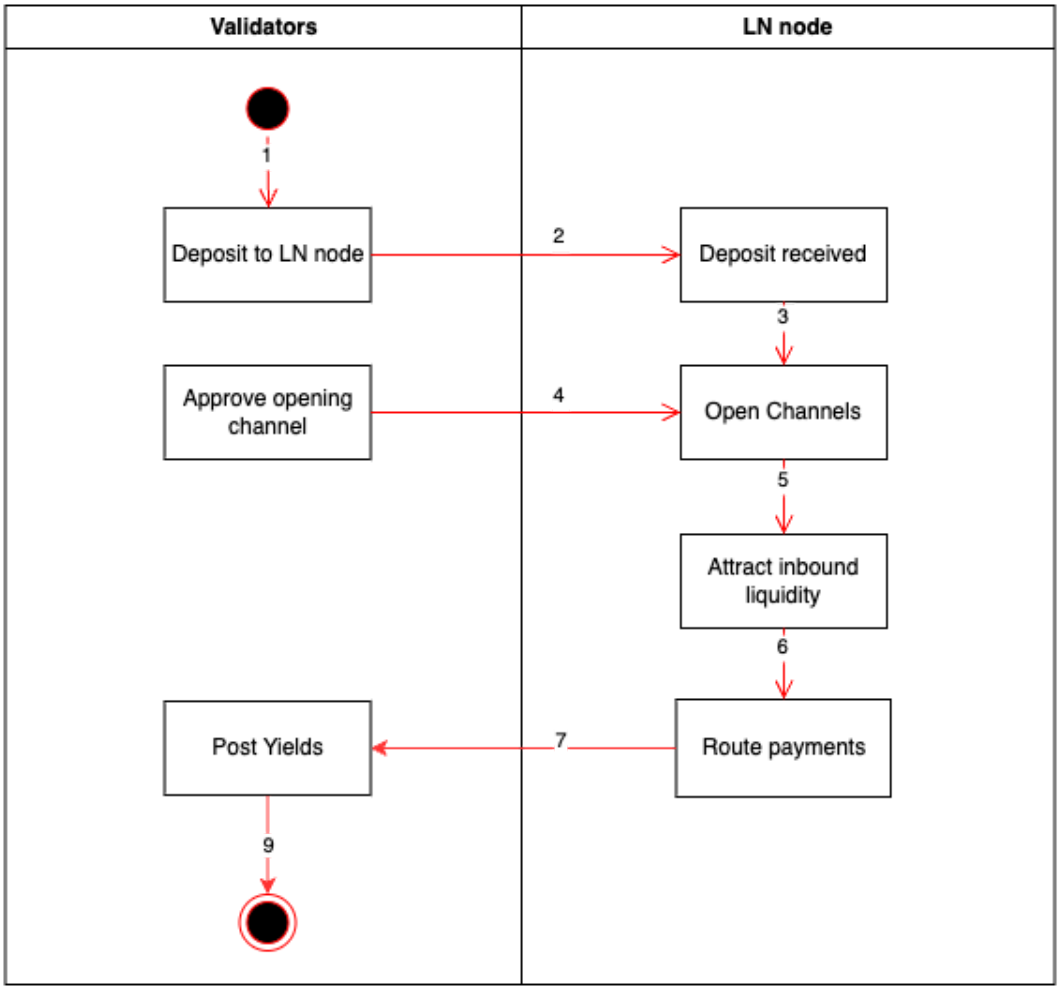

Diagram 8: LN Node Operations

The second promising opportunity related to the BitVM2 bridge is establishing a liquidity bridge [30] based on atomic swaps. Naturally, standard peg-out transactions through the BitVM2 bridge can be costly, lengthy, and inconvenient for users since only a specific amount of BTC can be pegged out. All of those limitations are implemented to ensure maximum BitVM2's bridge security. To make the bridge more usable, it is suitable to establish liquidity pools on both the

Bitcoin and bridged blockchain sides and do atomic swaps between those pools, making bridging transactions much faster, cheaper, and more convenient for users while preserving BitVM2's level of security.

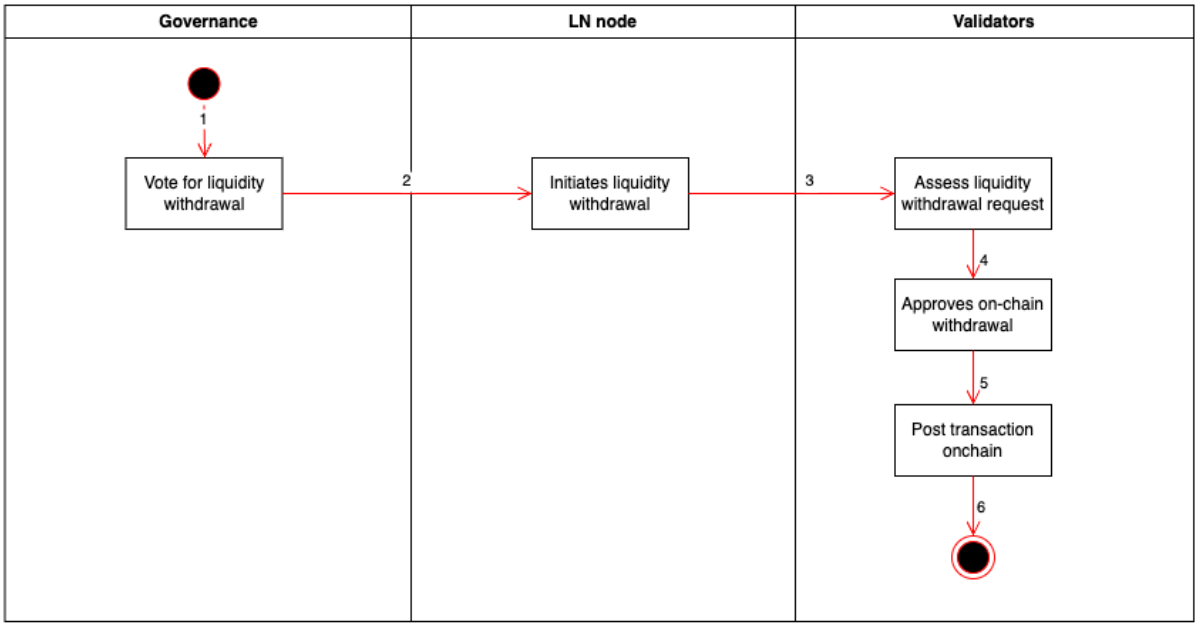

Diagram 9: Withdrawal from Lightning Node

While BitVM2 promises an unprecedented level of security for Bitcoin bridges, it is still in the R&D phase. Other BTC staking opportunities, based on different staking mechanics, are already gaining traction.

Bitcoin Restaking via Babylon

Bitcoin restaking through Babylon allows users to leverage their Bitcoin holdings to participate in securing Proof-of-Stake (PoS) networks without moving their assets off the Bitcoin blockchain. This integration enhances PoS network safeness in a capital-efficient way, allowing it to secure several chains with one stake. Rewards are usually paid in the PoS network token, which can be converted to Bitcoin and compounded to the Bitcoin pool. Lombard already proves that a federation-based bridge, which is staking with Babylon, can be implemented.

CoreDAO, Stacks, and Other Bitcoin L2s

In terms of the Stroom protocol, it is essential that it is possible to stake BTC in CoreDAO and Stacks consensus to receive revenue from protocol transaction fees. For example, as of February 2025, CoreDAO has around 5000 BTC staked in the non-custodial staking. Participating in their dual staking model can generate a 7.3% annual yield.

Whereas all of these earning opportunities look promising, one of the central questions related to the Stroom design is its security model. While we cannot employ the BitVM2 model to guarantee the security of the BTC locked in Stroom, we can leverage restaking as an additional security layer to improve Stroom's risk-reward ratio for its customers.

Economic Security Layer

Standard techniques, such as ensuring unique deposit addresses per user, sufficient block confirmations to prevent double spending, nonce protection from replay attacks, etc., cannot mitigate several fundamental attacks on the federation-based bridge. These attacks result from the bridge's reliance on a multisig federation, which is managed by several authorized parties. To better understand restaking utilization in Stroom, we first need to analyze potential threats to running a federation-based bridge.

Operator Risks

There are three vectors of attack on the federated bridges: native BTC theft, unbacked minting of the wrapped BTC, and censorship of the mints/redemptions.

In a federated model, control is distributed among multiple parties known as validators (usually a set of trusted signers). If an attacker compromises a supermajority of the federated nodes, they could gain control of the multisig wallet and initiate unauthorized withdrawals of BTC. However, it is not necessarily an external attacker who can steal the funds: validators can collude and steal funds for themselves.

While it is possible to restrict minting by requiring the minter to provide some sort of BTC deposit proof to the stroomBTC contract, either in the form of ZK proof or through chain relays [14], in practice, it heavily complicates the system. Moreover, it does not solve a fundamental security problem related to storing BTC in a multisig federation. Hence, in the Stroom protocol, the identical mechanism that is controlling the Bitcoin vault controls the minting process. Compromising this federated mechanism would allow the attacker(s) to mint an unlimited, unbacked number of stroomBTC on the Ethereum side, leading to insolvency in the Stroom protocol. stroomBTC uses the ERC-7265 [22] standard, aiming to limit the potential losses from the unbacked minting attack.

The third attack that can lead to users' loss of funds is a censorship attack. The user initiates the bridging transaction by sending the BTC to the deposit address or burning stroomBTC tokens. For transaction finalization, bridge operators must take some action: either authorize stroomBTC minting for the user or repay BTC to the redemption address. The operators can fail to perform their actions, leading to the user's funds being stuck in limbo.

Stroom utilizes restaking to minimize the risk of the attacks listed above.

Restaking

Stroom is using restaking protocol to make all listed attacks punishable for operators with economic losses substantial enough to significantly lower the potential benefit from performing an attack. Unlike employing a Stroom governance token only for that purpose, restaking allows the use of “hard” assets like ETH and stablecoin with deep liquidity to collateralize operators. Moreover, such collateralization can be done in a capital-efficient way since those protocols use LSTs like Lido’s stETH or eUSD and also allow them to be restaked to secure several networks simultaneously. As of Q1 2025, the amount of liquidity staked with restaking protocols is $25B.

Restaking pools securing the bridge can be viewed as insurance collateral. In case of a slashing event, the slashed funds are used as recovery reserves for everyone who has suffered from the attack on the bridge. On the slashing, slashed funds are moved to a special reclaim contract. Stroom protocol users will be compensated from that contract, fully or partially, depending on the restaking collateralization rate, which can be above or below 100%. The collateralization rate, in turn, depends on the amount of reward that is paid to the restakers.

Stroom protocol must adequately compensate restaking collateral for taking the slashing risks. That compensation is carried from the Stroom protocol’s revenue. Governance decides how much revenue is distributed to the bridge users and how much is paid to the restakers, depending on the desired risk-reward ratio for Stroom users.

Slashing

Anyone can initiate a slashing by supplying some financial bond and providing the slashing contract with proof of the unauthorized operation by validators. After that, stakes in restaking pools are frozen, and stroomBTC contracts are paused (excluding the case of censorship attack). The validators’ joint master key in the stroomBTC contract will be changed if needed. To mitigate some possible errors, after submission of the slashing event, there will be some grace period that operators can use to disprove the validity of the submitted slashing challenge. If the transaction the slashing initiator submitted is incorrect, proving it will be a straightforward process, and the challenger’s bond will be slashed. To incentivize challengers to perform their job of monitoring the bridge’s operations, some premium should be paid for a successful slashing submission, deducted from slashed funds.

In the case of BTC theft from the multisig wallet, it is an unauthorized transaction that happened in the Bitcoin blockchain. The challenger generates cryptographic proof demonstrating that the malicious transaction occurred in the Bitcoin blockchain and submits it in the slashing contract. This proof includes details like the block height, transaction hash, and other relevant data, like the existence or absence of redeem ID, that must be present in any transaction moving funds out of the Bitcoin multisig wallet. During the grace period, operators should provide proof of a corresponding redemption transaction made by a user, affirming the Bitcoin transaction was valid. The validity of cryptographic proofs is checked in the slashing contracts.

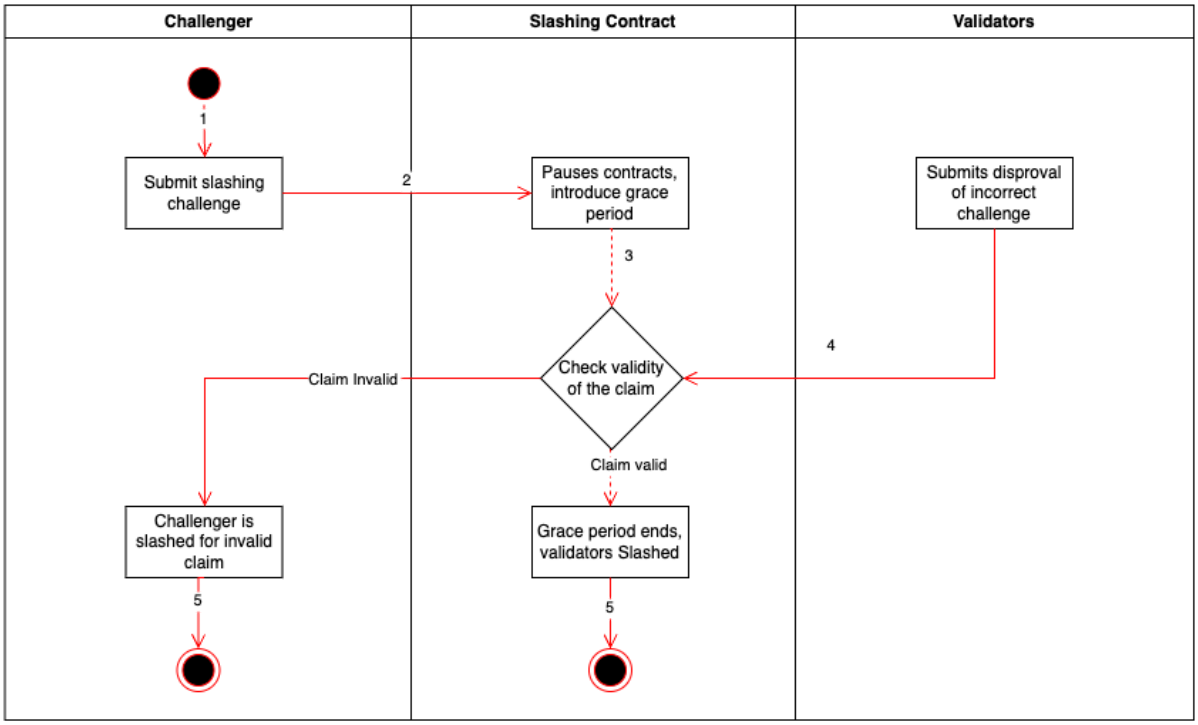

Diagram 10: Slashing Grace Perios

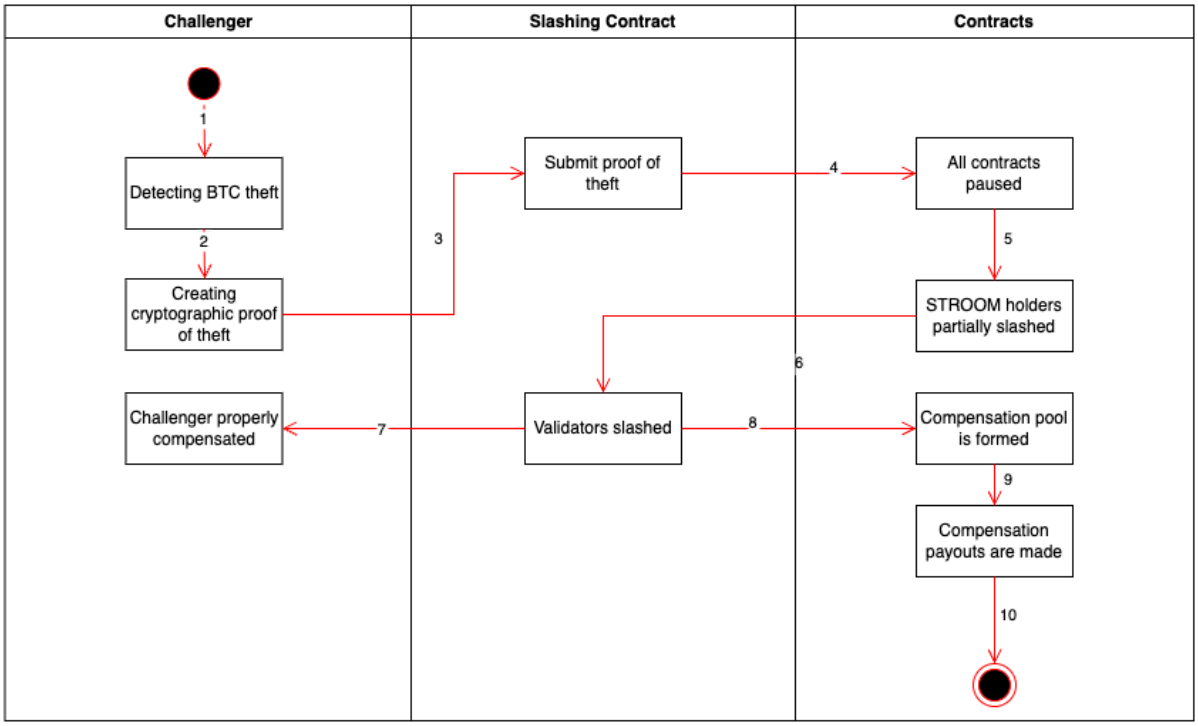

Diagram 11: Slashing When BTC Theft

If unbacked minting occurs, the challenger provides the details of the corresponding mint transaction on the Ethereum blockchain. During the grace period, operators should provide valid proof of the corresponding deposit transaction made by a user to Stroom's BTC multisig wallet. If such evidence cannot be submitted, then validators are slashed.

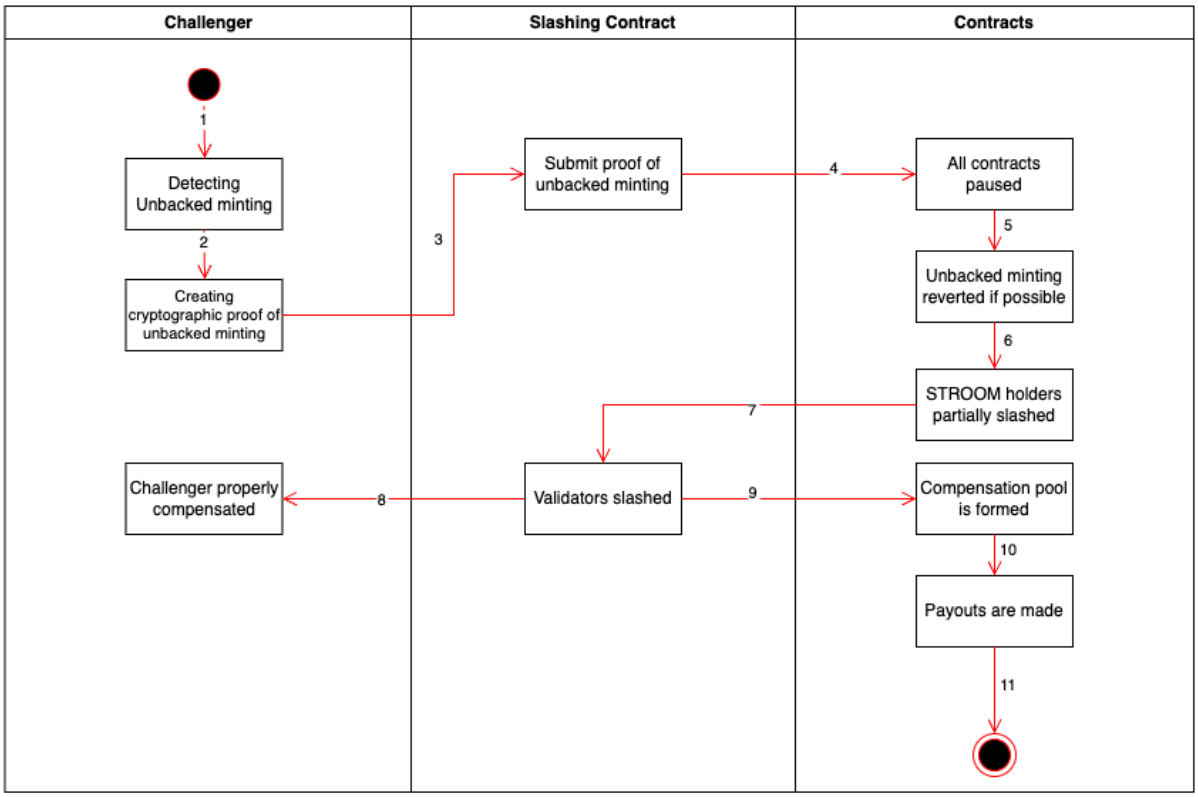

Diagram 12: Slashing When Unbacked Minting

When a mint or redemption transaction is censored, the affected user or any other challenger provides proof of the corresponding deposit on the Bitcoin blockchain or stroomBTC burning on the Ethereum blockchain. During the grace period, operators should submit proof that the corresponding mint authorization signature was created in case of the minting and proof of the BTC payout transaction in case of redemption. If such proof cannot be submitted, then validators are slashed.

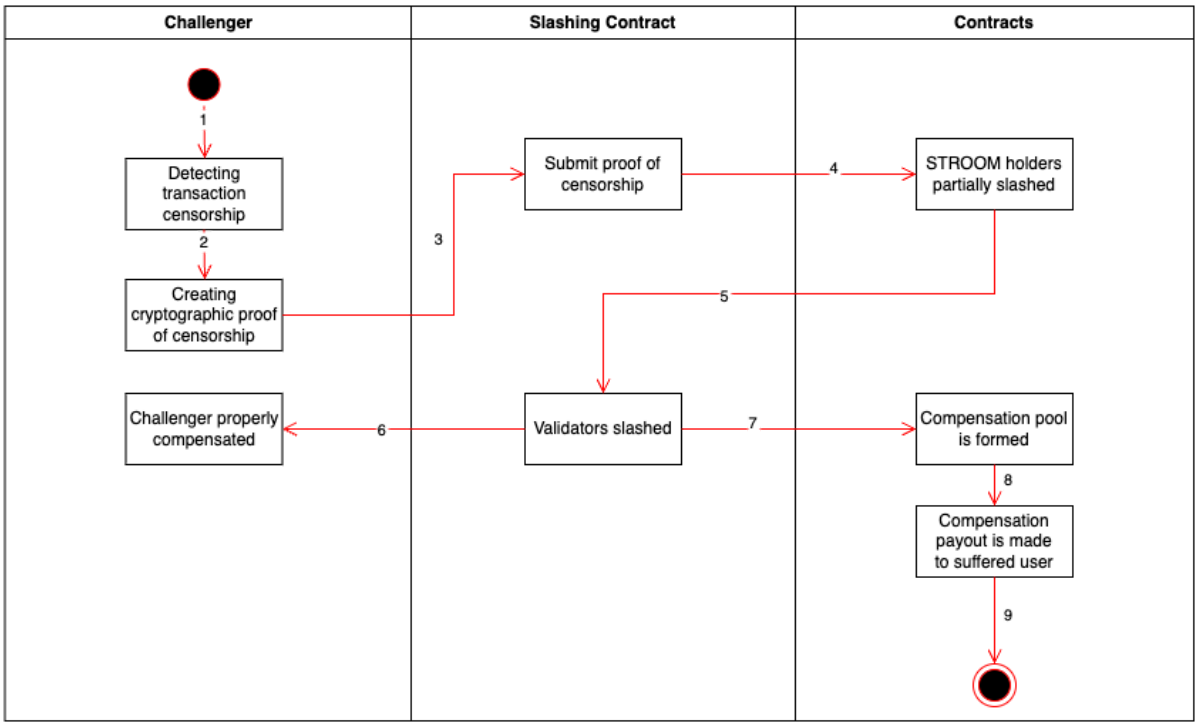

Diagram 13: Slashing When Censorship

Securing BTCs While Generating Yield

All routing LN nodes, BitVM2 operators, BitVM2 liquidity bridges, Babylon restaking nodes, and CoreDAO with Stacks validators within the system must adhere to the Stroom protocol and comply with its requirements. These nodes will not possess access to the BTC held within their respective vaults, and any action that modifies the vault state will require authorization from validating nodes.

A K-of-N Taproot multi-signature script will be employed to manage the hub's vaults instead of relying on a single private key, as is typically done in traditional setups. Notwithstanding, using FROST in the Stroom protocol allows the preservation of funds' custody within the consensus of validators the whole time while BTC liquidity generates the yield. With FROST, it is possible to fast-sign transactions for Lightning Network node operations or any other BTC liquidity provisioning activities if they are performed using Schnorr signatures.

In the case of Lightning Network, in addition to delegating private keys to the federation, it is mandatory for LN nodes to submit all revoked channel states, signed by the corresponding channel counterparty, to the databases of Stroom validating nodes. Consequently, the Stroom-enabled Lightning hub acts as a proxy for the federated Lightning Network Node, run by validating nodes. This configuration establishes an inherent Stroom Watchtower [31] functionality that automatically monitors the state of the channels. If an LN node fails to comply

with the rules established by the federation, it will take action by closing all channels associated with that non-compliant LN node and removing it from the system.

It is straightforward to implement restaking slashing conditions for yield-generation strategies on top of Babylon, BitVM2, and L2s since all operations of those strategies are performed on-chain. However, for conciseness purposes, we will not provide a detailed explanation of the implementation in this paper. Implementing slashing conditions for the Lightning Network is trickier since it is an off-chain protocol.

Two types of theft are possible in the LN node. First, closing channels and sending BTC on-chain to an address not controlled by the Stroom protocol. Since there is an on-chain trace for that action, proving this type of theft is fairly explicit.

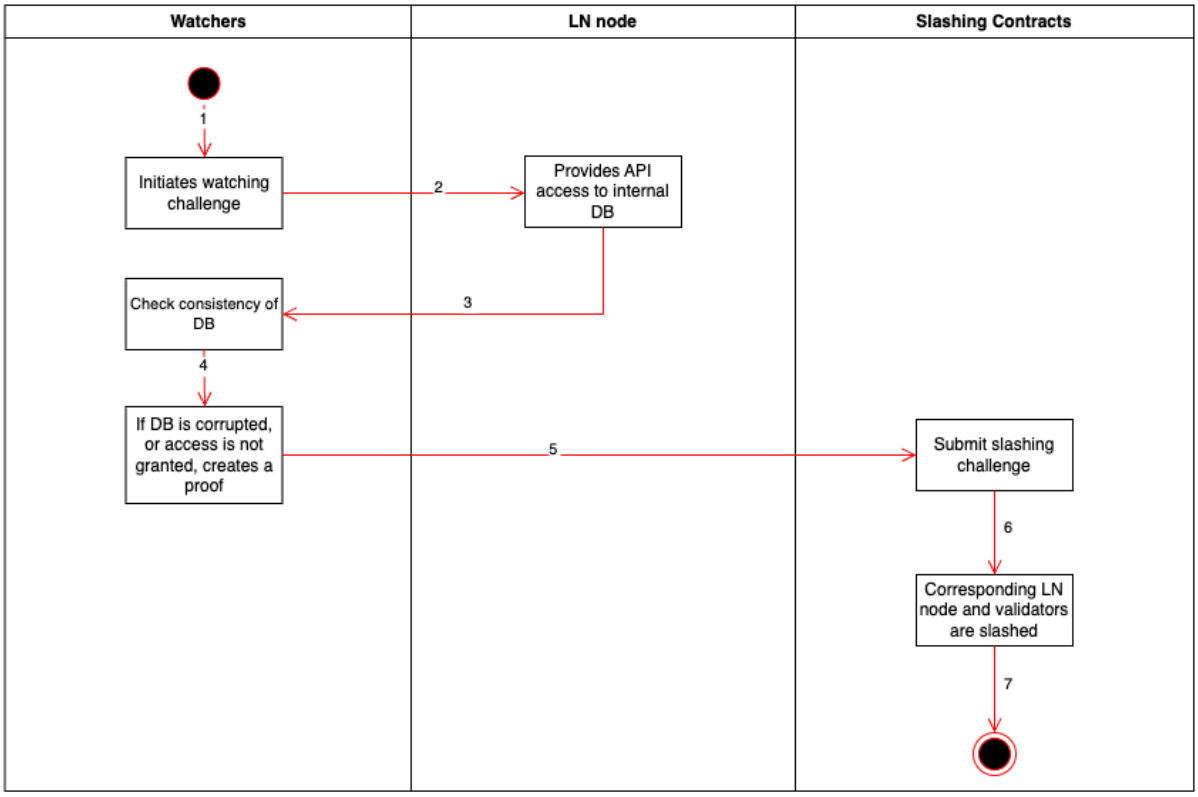

Diagram 14: Supervising LN Node Operations

The second type of theft is performed purely inside LN channels. During that attack, the thief sends off-chain LN payments to the node that is controlled by the attacker using the Lightning protocol. To confirm that no BTC theft occurred inside the Lightning Network node, the LN node operator must open read-only access to the node for the appointed watchers. Those watchers will periodically check the state of the Lightning Network node, ensuring that there was no BTC theft and auditing the rewards that were earned on this node. In case of a BTC theft or revoked access to reading data from the LN node, the watcher can provide a zero-knowledge proof of

that event to the slashing contract. The watcher will be rewarded with the percentage of the stake slashed for the proper slashing challenge. It is crucial that all watchers have access to the data from the LN node. Any of the watchers can submit the challenge. There is a grace period during which the LN node operator can disprove that the challenge was correct. The watchers can increase the strength of the check by sending a small payment through the LN node to be checked, and after that, they can query the node to see if that transaction is reflected in the node's database.

While such a scheme makes it possible to implement cryptoeconomically secured interest-bearing BTC with various sources of yield, the system must be configured to strike a proper risk-reward balance to be maximally attractive to capital. The DAO is responsible for finding such a balance.

DAO and Its Functions

The DAO's role is to curate key parameters of the Stroom protocol. The goal is to ensure that the four most important aspects of the bridge are functioning appropriately and adapting to the market conditions. Those aspects are:

- Maximizing protocol’s revenue
- Controlling the risks indirectly by incentivizing restaked liquidity
- Curating operators
- Properly configuring slashing

A DAO will be established by implementing a stakeholder token, STROOM. Locking the tokens allows DAO members to vote for key system parameters, e.g., how much liquidity to direct to the given BTC yield-generating strategy. Locking is implemented to guarantee voters’ accountability for their decisions. If the decision is flawed, locked tokens are going to be slashed. However, voters will receive adequate compensation for performing their duties successfully.

Selecting Sources of Revenue

Different sources of revenue must be adequately managed. The DAO decides how much liquidity to allocate to a particular liquidity provisioning method. The final goal is to maximize profit with the minimum risk possible. Liquidity allocation is voted on in the governance process, and the liquidity is allocated proportionally to the votes of the governance voters. However, voters are accountable for the choices they make. If the strategy has low performance, voters for that strategy receive fewer rewards. If strategy leads to loss of funds, voters are slashed, and their stakes are used to compensate the protocol for losses.

Controlling Restaking Collaterization Rate

The amount of restaked capital involved in the Stroom protocol can be below or above its TVL. If there is a significant surplus of staking collateral, the payouts can be decreased to save the revenue for the users, increasing yield for stroomBTC. On the other hand, lowering users' rewards and increasing the payouts to restakers attracts more stake if a greater level of security is needed. This provides a mechanism to find a proper balance for user rewards versus the protocol's security, which is most attractive to the capital. It is in the best interest of DAO to choose an appropriate rate of collateralization that is appealing to the users and attracts more TVL. More TVL will bring more revenue to the Stroom protocol and create more value for everyone involved.

The control of those rewards will be done by voting. Each STROOM token locked is accounted as one vote. Having some ballots, the DAO member can decide how to split those votes between directing rewards to the restakers or sending them to users. The aggregated decision across all voters proportionally to their total amount of votes will result in a final proportion of revenue distribution.

Managing Operators Set

Choosing proper operators is crucial for the safety of the Stroom protocol. STROOM stakers, as risk curators, will perform that task by voting. However, voters will be accountable for their votes. Voting for a validator with low availability will lead to lower rewards for the STROOM staker, whereas voting for a malicious validator who tries to steal funds from the protocol will lead to the voter's STROOM stake being slashed.

During the bootstrapping phase, the core development team will choose 100% of the operators. Later, gradually giving up control over validators to the DAO, the percentage of the core team's appointed operators will gradually drop to 0.

Configuring Slashing

Whereas slashing and compensating affected users in the case of a censorship attack is relatively straightforward, it is not that effortless in the case of BTC theft or unbacked minting.

After the unbacked minting, the core development team should burn or freeze tokens minted during the attack. However, a portion of unbacked stroomBTC may have entered DeFi liquidity pools and cannot be frozen or burned. From this point, BTC theft and unbacked minting are handled identically. That's because due to the BTC theft, there is some amount of unbacked stroomBTC in circulation, similar to unbacked minting. It could be that the total supply of stroomBTC is unbacked.

Restaking and stroomBTC contracts automatically pause, and slashing is performed. If some STROOM token holders voted for a slashed validator or the yield-generating strategy that led to

the loss, their STROOM tokens are confiscated and added to a compensation pool. Then, the staking pool is slashed, and all slashed funds are directed to the compensation pool.

The final goal is to resolve the insolvency of the protocol using available collateral in the compensation pool. If it is clear that the compensation pool is significantly lower than the value of stroomBTC needed to be compensated, then there are two options. STROOM holders can dilute their STROOM token holdings by slashing a portion (e.g., 30%) of their token holdings to the compensation pool. If there are still not enough token funds in the compensation pool, then there is a collateralization of last resort - issuing a debt token that STROOM protocol will be paying out gradually from the revenue of the protocol. Naturally, all BTC that is left in the STROOM protocol and collateral in the compensation pool will be issued pro rata to the token stroomBTC holders, which, as a result, will buy back stroomBTC from the market. The remaining stroomBTC will be converted to debt tokens.

The auction will be set up if there is enough collateral to cover losses. This process involves the auction to rebuy the unbacked stroomBTC from the market using assets from the compensation pool. Users who want guaranteed compensation for their stroomBTC tokens can opt into an auction at a lower price but with guaranteed recovery.

Future Developments

The Stroom protocol will be launched in several phases. During the initial bootstrapping phase, Stroom will involve a closed group of authorized custodians managing Lightning Network nodes as the first source of revenue and a diverse group of validating nodes maintaining the bridging platform chosen by the DAO governance process. Subsequently, in the following phase, Stroom will undergo a transition towards a decentralized autonomous organization - DAO-governed yield-generating vaults, including yields from Lightning Network, Babylon, BitVM2, and CoreDAO with Stacks. DAO members will always govern protocol control and development through DAO voting. In the following phases, Stroom will add support bridging to the other chains like Solana, Ethereum L2s, Bitcoin L2s, etc.

An essential direction of the protocol development is integration with the other Bitcoin protocols that require BTC liquidity, similar to Lightning Network and BitVM2. That can be any Bitcoin L2 that requires Bitcoins to secure operations, Bitcoin staking protocols, etc. Such integrations can unlock new liquidity provisioning sources of revenue for the Stroom protocol.

Cross-chain messaging can be further improved by introducing chain relays, which will make unbacked minting attacks impossible. Furthermore, proper rotation of validators can improve the censorship resistance of the Stroom protocol.

Finally, the Stroom protocol aims to become a bridge and a consensus-as-a-service layer for any Bitcoin application. Using Stroom’s security layer, developers can focus on building a product without worrying about securing Bitcoin vaults. Such applications include anything from a Bitcoin-backed stablecoin to a full-scale L2 chain or a Bitcoin-native lending market.

References

[1] Babylon Whitepaper
[2] Berachain Docs
[3] Bitcoin Bridges by DefiLama
[4] Bitcoin Bridges Summary
[5] Bitcoin Transaction Sequence Number - from the Book
[6] Bitcoin Whitepaper
[7] Bitlayer Bridge to Ethereum blog
[8] Bitlayer Whitepaper
[9] BitVM2 Paper
[10] BitVM2 Summary
[11] Brontide Protocol
[12] BTCB Contract
[13] Ceffu
[14] Chain Relays
[15] Cobo
[16] CoreDAO Whitepaper
[17] Cryptoeconomics Wikipedia
[18] DeFi Summary
[19] Eigenlayer Whitepapers
[20] ERC-20 Standard
[21] ERC-4626 Standard
[22] ERC-7265 Standard
[23] Ethena
[24] Ethereum Whitepaper
[25] FROST
[26] Jupiter Perpetual DEX
[27] Lightning Adds USDT
[28] Lightning Network Whitepaper
[29] Lightning Network Yield by AMBOSS
[30] Liquidity Bridge - Blog
[31] LN Watchtower
[32] Lombard.Finance
[33] pBFT Summary
[34] Proof-of-Stake Wikipedia
[35] RenBTC to shutdown
[36] Restaking Summary
[37] River Report
[38] Schnorr Signature Wikipedia
[39] Solv Protocol
[40] Stacks Whitepaper
[41] Symbiotic Docs
[42] Taproot Bitcoin Addresses - Blog

[43] tBTC
[44] Threshold Network
[45] Understanding Slashing Conditions
[46] VLS Tech
[47] WBTC