L15 Trust Model

Securing a Trustless Side Chain for a Bitcoin-Backed Stable Coin

Miro Sarbaev Lexis Tikhvinskiy

August 9, 2022
Commit b56df75

Abstract

Layer One-dot-Five (L15) is a variant of a trustless sidechain to Bitcoin that enables BTC-backed loans in a stable coin. The paper shows how L15 maintains decentralized trust, focusing on a bridge between the sidechain and Bitcoin, where security is enforced via a combination of two approaches: 1. Focusing on *deterministic use cases* where the outcomes are known in advance, allowing for the wide use of DLC. 2. VSD / VC-P (Validated Speck of Dust / Virtual Counter-party) method, secured by L15 merge miners, who are aware of the sidechain consensus rules and mine Bitcoin blocks; miners are supplemented with the functional extension of validators, who provide them with the ability to create persistent signatures and to store the contract data.

For the latest version of this paper please visit
https://L15.dev/L15-trustmodel.pdf

Acknowledgments

The authors would like to extend their sincere thanks to:

Dr. A. Bekin, co-author of the *L15 Economics* paper for significant contributions to making the paper digestible.

Katherine N. Golubev, the communications artist, who made the English language in this paper flow.

K., for providing the Bitcoin industry guidance and bringing the spirit of the true cypherpunk into the mix.

Contents

1 Introduction

A stablecoin that one can borrow against the value of their Bitcoin is a sound approach to increasing the ecosystem's money velocity. While an ERC20-based solution is available to Bitcoin holders via WBTC, most users find this path too expensive and scary.

Users' fear is justified when considering the compounded risk of several innovative solutions chained together and the added exposure to centralization risks introduced by some of the intermediaries. The cost of time and attention needed to traverse these layers, as well as the high fees, when put together, make the process more trouble than it's worth. Fundamentally, these are all issues of trust, along with the fees. If a collateralized loan solution was built with a decentralized trust mechanism that does not depart too far from the one of Bitcoin, while presenting acceptable tradeoffs, it could establish continuity with the ecosystem and implement market-formed fees. The interconnection between trust and value becomes even more apparent if we remember that Bitcoin can be defined as a provider of decentralized trust.

The ultimate goal of L15 (Layer One-dot-Five) is to address the challenges listed above. It enables L15$ - a Bitcoin-backed stablecoin, near-native to the Bitcoin ecosystem. To support the claim, L15 must be able to uphold a degree of trustlessness and decentralization nearing the Bitcoin standard and validate its trust model, which is the main focus of the present paper.

2 Defining the Term "Trust Model"

"Generally an entity can be said to 'trust' a second entity when the first entity makes the assumption that the second entity will behave exactly as the first entity expects" X.509 ITU Standard (ITU, March, 2000) Section 3.3.54*

With that in mind, we define a *Trust Model* as a description of processes, entities, and interactions created to assist predictions about a system behaving as expected.

Trust is (*Trust Modeling for Security Architecture Development*, Apr 18, 2003): "a balancing of liability and due diligence" and "a binary relationship or set of compounded binary relationships". Combining the two characteristics, ongoing trust can be described as a binary *go/no-go* decision based on a compounded dataset of relationships, where each relationship falls somewhere between the two extremes. The article aims to present the L15 trust model and show that its internal co-relations support a positive prediction of behavior.

3 Expected Behavior

Stablecoin L15$ is a product of overcollateralized loan facilitated by the L15 contract. Users make deposits in Bitcoin and borrow against the deposited value. Repaying the loan is equivalent to providing proof of burn for borrowed L15$ and accrued interest, which automatically makes the Bitcoin deposit spendable by the borrower. If the value of the Bitcoin deposit falls below the minimum collateralization ratio, the loan can be liquidated by the contract. The US dollar value of L15$ is backed by the US dollar value of the underlying Bitcoin deposits held by L15 contracts. To maintain the L15$ price equal to USD, L15 regulates fees and loan interest rates and issues or burns the native Stabilization Reserve coin - L15SR, which is also used to reward miners and pay most transaction fees.

4 Trustless Sidechain Variant

The starting point for L15 design is the concept of a pegged sidechain (Back et al., 2014, October 22), where instead of an asset that 1:1 represents locked Bitcoin, L15 issues a stablecoin L15$ backed by the locked Bitcoin, as shown in fig. 1.

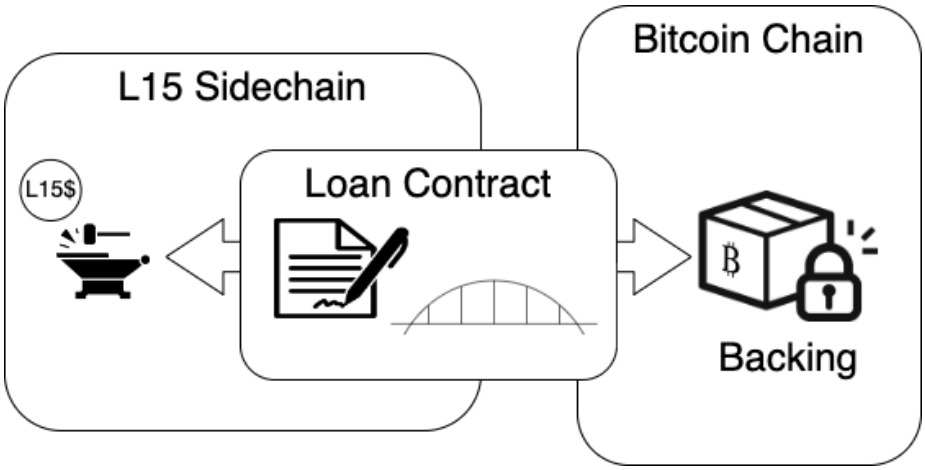

Figure 1: L15 as a Sidechain

In order for L15 to utilize *distributed consensus*[1]across the board, two challenges must be resolved. These challenges are well-documented, in large part by their own creators — the *small-cap chain challenge* results in an imbalance with a game-theoretical incentive for the members of *distributed consensus* to destroy the chain[2], and the *expressiveness challenge* refers to Bitcoins' inability to look outside of itself [3]. These challenges are usually mitigated by "having a trusted *federation* of mutually distrusting *functionaries*"[BCD+14].

In order to stay in the realm of identity-lacking *distributed consensus*, L15 limits

[1]We use the term *distributed consensus*, meaning *"a consensus (i.e., global agreement) between many mutually-distrusting parties who lack identities and were not necessarily present at the time of system set up."*(Poelstra, 2015, March 22)

[2]probable condition that occurs when the value of Bitcoin controlled by the sidechain's distributed consensus is higher than the sidechain capitalization.

[3]The *expressiveness challenge* is described by the inventors of pegged sidechains (Back et al., 2014, October 22): "One of the challenges in deploying pegged sidechains is that Bitcoin script is currently not expressive enough to encode the verification rules for an SPV proof." In practical terms, it means that Bitcoin does not have any internal instruments that can make it aware of transactions in a "chain B". As a result, if one wants to create a transaction in Bitcoin that is triggered by an event in "chain B", it must rely on an outside entity. If there is a goal to create a bridge from "chain B" to Bitcoin and keep it trustless and decentralized, the resulting tandem will automatically be less secure than Bitcoin. Using a centralized authority or federation is a good way to secure the bridge at the expense of trustlessness.

itself to supporting what we define as *deterministic use cases* where the outcomes of the contracts are known in advance and Bitcoin is cryptographically locked to these outcomes. [4] Furthermore, it implements a Validated Speck of Dust / Virtual Counter-party (VSD/VC-P) method, a layered inter-relation of entities, described below. The introduction of these two new solutions working together provides reasonable security for the system's operations.

We will start the detailed review with the latter: VSD/VC-P. After that, we will focus on L15 functionality: how limitations of the use case of collateralized lending protect the areas that otherwise would have become vulnerabilities.

5 VSD/VC-P Method

VSD/VC-P is based on the use of specifics of the underlying sidechain consensus; L15 is a PoW chain that uses a familiar algorithm, merged mining with Bitcoin:

> If an L15 miner solves a Bitcoin block at Bitcoin difficulty, he will commit the block to the Bitcoin chain with an anchor - L15 hash of the block being mined. Since Bitcoin block difficulty is greater than that of L15, a solved block in Bitcoin also counts as an L15 block solution. Conversely, if an L15 miner finds a solution at the L15 difficulty first, they record the contents of the Bitcoin block inside the L15 chain as proof of their work, which counts as an L15 block solution.

We will be referring to an L15 miner who solved a block at the Bitcoin difficulty as *L15minerB*. Respectively, an L15 miner who solved a block at the L15 difficulty is called *L15minerL*.

[4] Strictly speaking, a collateralized loan contract cannot be considered a *deterministic use case* in its totality. For example, in the case of liquidation via an auction, it is unknown who will end up owning it in advance. Exceptions like that give the distributed consensus control over fractions of deposited Bitcoin for a limited time under specific circumstances. The existence of such limitations dramatically reduces the probability of misappropriation of users' deposits. The remaining low probability is mitigated on a case-by-case basis.

5.1 Use Case

L15 uses VSD/VC-P to enforce a consensus rule that a transaction $PTrx$ must appear in the Bitcoin chain only if the L15 chain contains a record of $LEvent$, as shown in fig. 2.

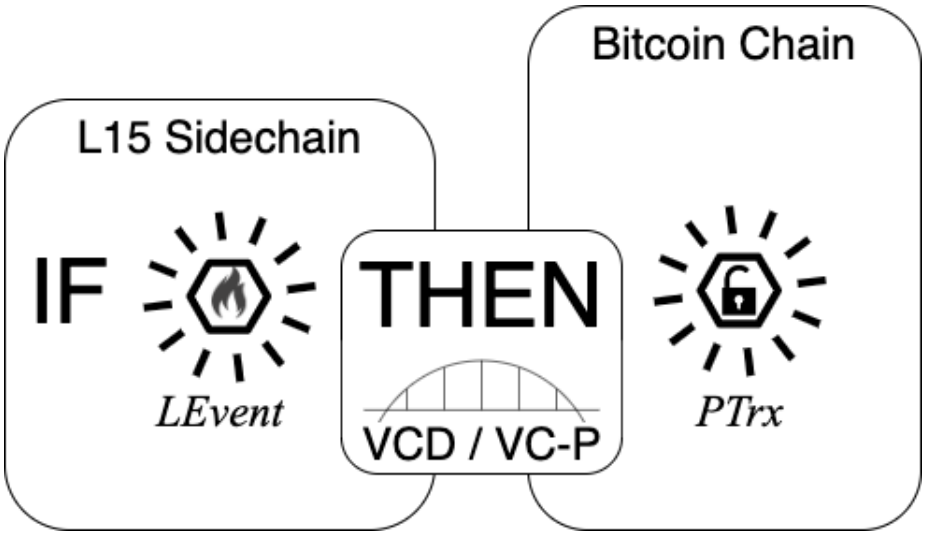

Figure 2: VSD / VC-P Usecase

Since it is a *deterministic use case*, un-actualized $PTrx$ must exist prior to the $LEvent$ in the form of a script that spends a pre-defined Bitcoin output to a pre-defined address. Before $LEvent$, $PTrx$ is a Bitcoin Layer 1 spending script, similar to a Bitcoin Layer 2 payment channel commitment. In Lightning, commitments are off-chain; conversely, L15 records them in Layer 1 of its own chain.

5.2 Miners and the Validated Speck of Dust (VSD)

Staying within the realm of *distributed consensus*, the decision to actualize $PTrx$ must be carried out by miners. In L15, this role is fulfilled by an *L15minerB*: he is already enforcing the consensus rules of L15, and after winning a Bitcoin block, he has the power to create transactions in Bitcoin Layer 1. *L15minerB* verifies the presence of *LEvent* in the L15 chain, which, according to the L15 consensus rules, warrants the legitimacy of inclusion of $PTrx$ in the Bitcoin block.

L15 must be able to enforce the *exclusivity rule*: only an *L15minerB* and no one else can include $PTrx$ into a Bitcoin block.

To ensure this, $PTrx$ has a one-satoshi output, which makes it inadmissible to the Bitcoin mempool due to the dust limit. At the same time, a transaction with a one-satoshi output is legal from the standpoint of Bitcoin consensus rules and can be included in a block by miners.

In practice, a working limitation by a one-satoshi output requires a minimum of two consecutive *L15minerB*s, with an enforced maximum time allowed between the actions by $L15minerB_{(1)}$ and $L15minerB_{(2)}$. [5]In consecutive order, these miners are referred to as "The Publishing Miner" $L15minerB_p$, and the subsequent n miner(s) as "The Validating Miner" $L15minerB_{v(1...n)}$. The sequence can be further strengthened by increasing the number of consecutive *L15minerB*s, which comes at the expense of the completion time. This process increases the security of the cross-chain transaction to a level that approaches what one would expect from a PoW.

However, while this mitigates the chances of attack from within the chain, a one-satoshi output enforces the *exclusivity rule* only in part: $PTrx$ can still be actualized by a Bitcoin miner. This makes L15 vulnerable to attacks by Bitcoin-only miners who may not have any direct gain from the attack's success but at the same time will have nothing to lose either; an attacking miner can mine a release transaction bypassing validation steps and still get his Bitcoin block reward. This attack becomes especially relevant if an attacker is associated with one of the large mining pools.[6]

[5]Otherwise, a compromised L15 Miner with a false $PTrx$ can simply wait until he wins the block and becomes an $L15minerB_{(1)}$, in order to include $PTrx$ with certainty. This kind of attack is mitigated by bringing in an additional miner, making it necessary for the attacker to guess $L15minerB_{(2)}$ before the end of the time limit.

[6]The possibility of an attack is not completely mitigated by the new version of Stratum: the v2 of the pool protocol refocuses the attack from the mining pool to their block template providers but keeps the attack surface uncovered.

5.3 Virtual Counter-Party (VC-P)

L15 protects itself from Bitcoin-only miners and mining pool attacks by requiring a validating signature from L15 on any $PTrx$, i.e., making L15 a signing *counter-party* to any contract outcome.

A *counter-party* must be able to store specific contract-related data and respond to changes in the contract state. In other words, the *counter-party* must be able to have secrets and be highly available. PoW is not capable of either due to its stateless nature: miners come and go randomly.

Signers

To functionalize the *counter-party* requirements, L15 introduces a new type of system participant: a signer. Signers cannot be used individually[7], they are assembled into collectives - the L15vroutnodes. When a user opens a new contract, a dedicated L15vroutnode is assembled out of the pool of all available signers.

L15vroutenodes are not the *counter-parties* per se: they are designed to be "service providers", delivering services of a per-contract persistent memory and a "signing hand." The *counter-party* to all the L15 contracts is the whole L15 system.

Bonds and Assemblage

Signers are required to supply bonds in Bitcoin, which is helpful in many ways, not least to help prevent Sybil attacks. The selection of a signer for an L15vroutenode is probabilistically random, with the random component provided by entropy from the L15 miners and the probability depending on the total size of the bond.

Signers lock their bonds and get paid per signature. Essentially, it is the equivalent of signers "parking" their bitcoin and collecting interest on it, given that they are present to provide a signature. The size of their bond is equivalent to the interest rate: they are market-driven, based on availability to sign.

[7] if they were, the secrecy requirement would contradict the ground rule of participant's anonymity as one cannot make assumptions about the abilities of an anonymous signer to keep secrets. Likewise, the availability of an individual signer for a one-year duration of a typical loan contract is also problematic. The solution is to assemble a collective signer - the L15vroutnode, a "Flying Macaroni Monster" consisting of individual "noodles" – signers.

Threshold Multisig and Communication via L15 Chain

An assembled L15vroutenode can produce a signature with a private key that individual participants do not know by using Schnorr threshold multisig to build collective counter-party signatures.

When Alice transacts with L15vroutenode using a payment channel[8], it is initiated as a 2-of-2 multisig on Bitcoin Layer 1. The first signature is Alice's ($sig_{(\text{Alice})}$), and the second is the threshold multisig of L15vroutenode ($sig_{(\text{L15vroutnode(i)})}$), as shown in Equation 1, referred to as the (K-of-N)+1 signature scheme:

$$A = sig_{(\text{Alice})} + sig_{(\text{L15vroutnode(i)})} =$$

$$= \underbrace{sig_{(\text{Alice})} + \overbrace{(sig_{(\text{signer 1})} + sig_{(\text{signer 2})} + ... + sig_{(\text{signer k})})}^{\text{K-of-N}}}_{\text{2-of-2}} \tag{1}$$

This allows:

1. Alice to communicate with L15Vroutenode as a single entity,
2. Signers to interact with each other and produce threshold signatures for the state channel with Alice, and
3. Merged miners to validate these communications and enforce L15 consensus rules.

Once the funding transaction in Bitcoin is created and the payment channel between Alice and L15Vroutenode is established, the counter-parties communicate the state changes to each other in the channel using a protocol similar to BOLT(*BOLT V3, Bitcoin Transaction and Script Formats*, n.d.). From the standpoint of Bitcoin, these are lightning-like Bitcoin Layer 2 commitments. On the L15 side, these commitments are recorded onto the L15 chain Layer 1. In order to avoid recording traceable HTLC preimage and payment hash on-chain, L15 uses PTLC instead of HTLC.[9]

While several threshold multisig implementations are associated with identified security issues [10], L15 has a "natural immunity" to these vulnerabilities because

[8] As defined in https://en.bitcoin.it/wiki/Payment_channels

[9] In a departure from the Lightning network's protocol, recording the commitments in a public blockchain requires a modification in the way commitments are formed. HTLC preimage and payment hash must be private, which is not a problem for the Lightning network since communications between its peers are direct. In the case of L15, if HTLC is used, transaction data would be made public by recording it on-chain. This data would include payment hash and preimage, which would be unacceptable.

[10] in the context of OMDL (Drijvers et al., 2018)

the attacker must be capable of spamming L15vroutenode with signature requests. In L15 calls are recorded in the blockchain. This means that the process is slowed down — requests cannot be issued in parallel and Alice must pay per-record fees. Moreover, since there is a record, attacks can be identified and mitigated.[11]

5.4 The Undo

The sequence of steps we have described so far consists of a validation by the L15vroutenode and by two or more consecutive *L15minerB*s. If we compare that to Bitcoin and set aside the difference in hash power, the described "chain of command" provides reasonable assurance that if a $PTrx$ must be included in the Bitcoin chain, it will be included.

However, there is no such certainty in the opposite direction. If an adversary mounts an attack that results in $L15minerB_{(2)}$ including an illegitimate $PTrx$ in Bitcoin Layer 1, L15 has no recourse so far.[12] To remedy this, L15 has an additional entity in charge of an "Undo Button": the *Majordomo*.

Majordomo is a collective signer, assembled in a similar way to L15vroutenode, but out of a pool of different signers: they are required to supply bonds in L15SR rather than in Bitcoin. The realm of the *Majordomo* is limited to undoing $PTrx$ transactions once they have passed the stage of the Publishing Miner, and returning $PTrx$ back into the contract. They are also in charge of handling the edge cases [13] In such cases, they are responsible for identifying statistical outliers and attacks, and assigning fees to appropriate parties.

Majordomo acts within a time window T_{delta}. Within this time frame, the UTXO from a transaction recorded by the Publishing Miner becomes spendable either by the transaction that will be recorded by the Validating Miner after a relative time delay T1 or by a rollback transaction by the Majordomo after a relative time delay T2, where T1 > T2. Both T1 and T2 allow to clear potential normal reorgs in the L15 chain, but the time gap between T1 and T2 creates a possibility to roll back the transaction before the proceeds of $PTrx$ are sent back to Alice. *Majordomo's* time window is defined as T_{delta}= *T1 - T2*.

5.5 Loan Payoff via VSD/VC-P

In the case of trustless loan payoff, the use case described above can be restated: L15 uses VSD/VC-P to enforce a consensus rule that a collateral release trans-

[11] Additionally, (K-of-N)+1 can be constructed using FROST (Komlo & Goldberg, 2020, December 22), the algorithm designed to resist the said class of attacks.

[12] A similar situation could emerge as a result of a legitimate reorg of the L15 chain.

[13] like ones where the Validating Miner has not shown up on time, which is statistically possible in the course of L15's normal operation.

action ($PTrx$) must appear in the Bitcoin chain only if the L15 chain contains a record of an appropriate burn of L15$($LEvent$). For that, as shown in fig. 3:

1. Alice creates a Burn Request in the L15 chain and commits the payoff in L15$ to it.
2. L15Vroutenode signs the burn request, revealing the key to release the collateral to Alice.

3. Now all the data that is required to release the collateral is waiting for the Publishing Miner as a record in the L15 sidechain.
4. Publishing Miner picks the data, forms the first stage of release transaction, includes it into his Bitcoin block candidate, and mines it.
5. Now the collateral UTXO is waiting for the Validating Miner. Up until the time of T2, Validating Miner is the only entity that could do anything with the UTXO. After T2 but before T1 Majordomo can step in and rollback the transaction, together with the burn. After T1 release transaction is canceled and collateral UTXO again becomes spendable by the loan contract. Details of the rollback, including Flash Payoff will be discussed in the later section of the paper.
6. Validating Miner mines the second stage and now UTXO becomes spendable by Alice. Steps 5 and 6 can be repeated.

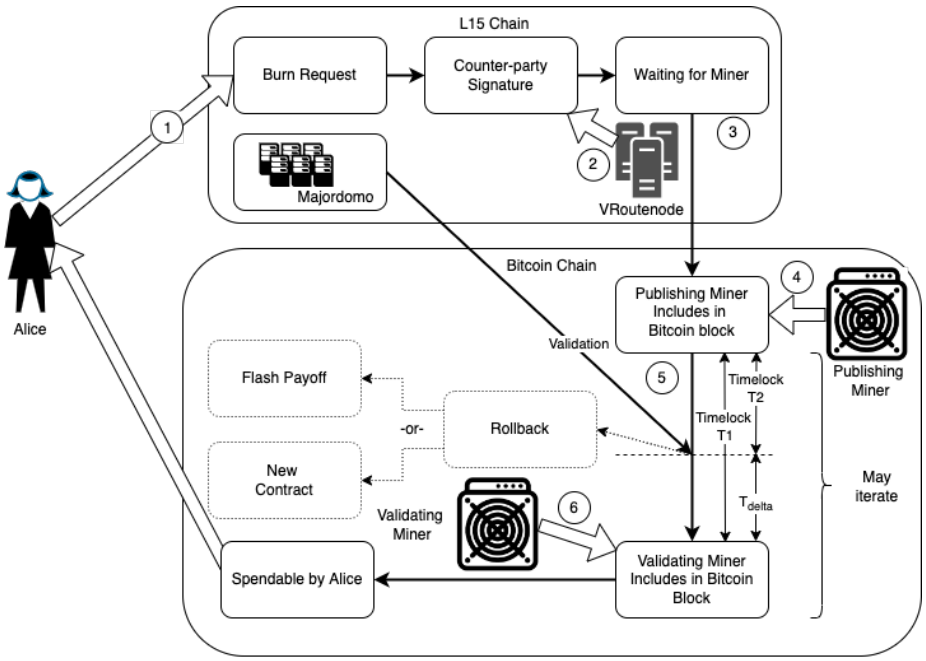

Figure 3: VSD / VCp

5.6 On Stake Without Consensus

An arrangement with signers is reminiscent of a Proof of Stake algorithm. However, an essential difference precludes L15 from being impacted by the PoS trust model deficiencies described in (Poelstra, 2015, March 22). The fact that signers do not participate in recording or maintaining the history of the L15 blockchain separates them from the power that would allow them to create a vulnerability.

6 Functional Analysis

When viewed from a functionality angle, the L15 trust model forms a hierarchy of binary relationships, as shown in fig. 4.

On the top level of the tree:

1. The system is trusted to be a timestamp server, just like Bitcoin (fig. 4), L15 blockchain aims at a 60-second block time. Merged mining provides a natural way of recording L15 anchors into the Bitcoin chain, which eases the timekeeping responsibilities but also increases complexity because it creates requirements for the synchronization of chain clocks while considering reorgs.

2. The system is trusted to maintain price stability of L15$ (fig. 4, 2). This function is implemented entirely in the sidechain, depending on the quality of the PoW consensus. Additionally, it has to trust the pricing oracles, Olivia(s), who publish the BTCUSD ratio. The oracles (Guillyr et al., 2020, November 03) and multi-oracle support (Cohen, 2021, May 07) are assumed to conform to Suredbits specifications.

3. The system is trusted to work with loan contracts (fig. 4, 3). This part of functionality contains most of the challenges and innovations of the L15 system.

6.1 Loan Funding

Expectation: At this step, L15 ensures that if Alice posts collateral of X Bitcoin, she will walk away with Y Amount of stablecoin L15$, as per (fig. 4, 4).

Execution: Alice (fig. 5, 1), sends X Bitcoin into a cross-chain atomic swap (fig. 5, 2) and swaps[14] her Bitcoin for Y L15$. The counter-party to this transaction is

[14]We are referring to both L15$ and L15SR as both being native assets on the L15 chain because this is how they are implemented. There are tradeoffs between a multi-asset chain implementation vs a single-asset chain with the second asset tracked via a solution similar to a color coin. From the standpoint of the present paper, the tradeoffs and cost of switching to an

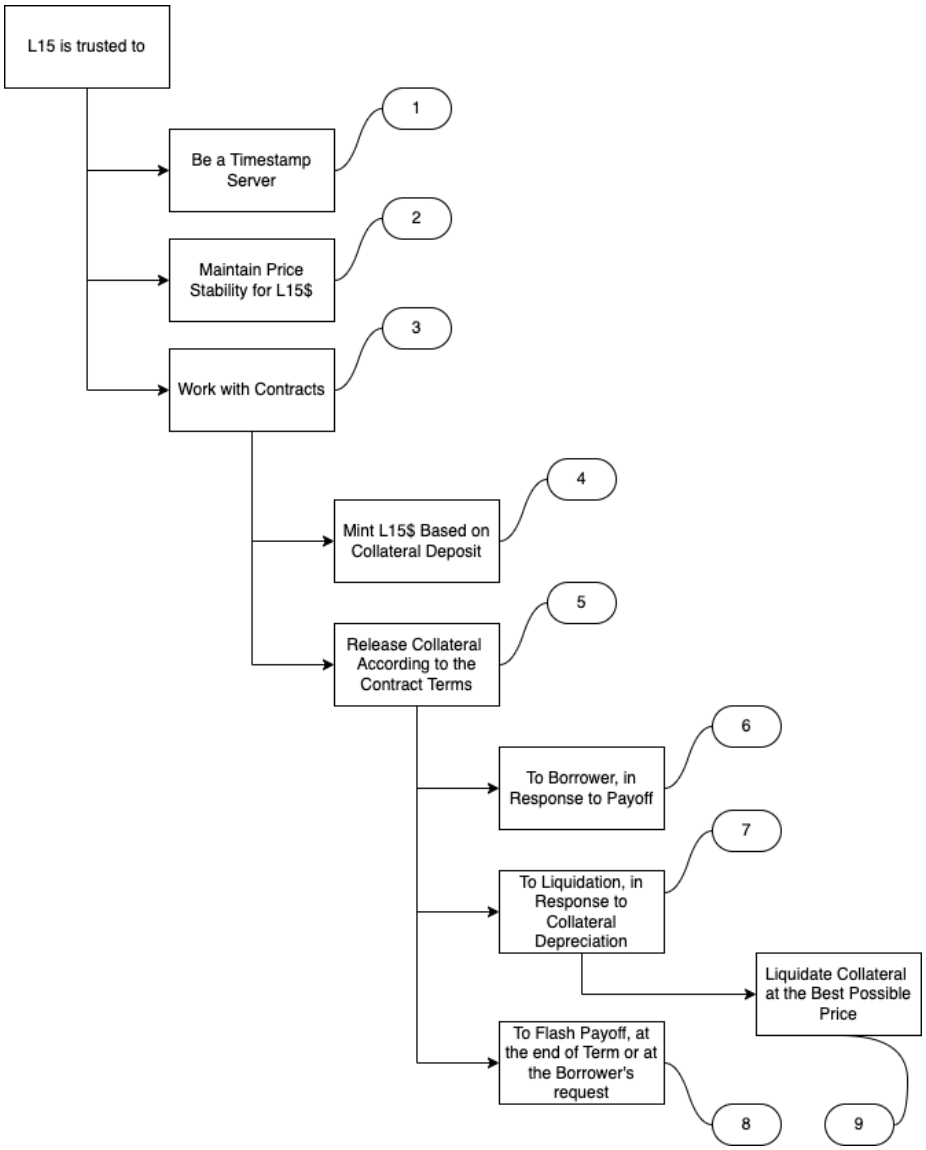

Figure 4: Compounding Hierarchy of Binary Trust Relationships

L15vroutenode$_{(A)}$ (fig. 5, 3), which funds its part of the atomic swap by minting the appropriate amount of L15$ on Layer 1 of the L15 chain and sending it into the swap. As a result, Alice walks away with the expected amount of the stable coin Y L15$, spendable by her key in the L15 chain.

Security: This step is maintained cryptographically; the participating parties validate the contract scripts.

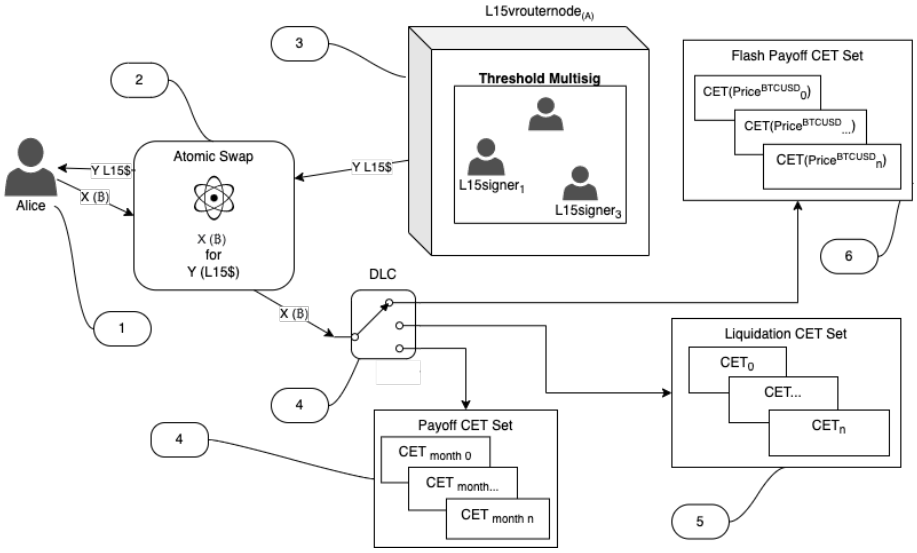

Figure 5: Loan Contract: Atomic Swap and DLC

6.2 Storing and Releasing Collateral

Successful atomic swap makes Alice's collateral spendable by the Contract DLC[15] script(fig. 5, 4 and fig. 4, 5), which has three groups of CETs[16]:

- Payoff set of CETs (fig. 5, 5)
- Liquidation[17] set of CETs (fig. 5, 6)
- Flash Payoff[18] set of CETs (fig. 5, 7)

When stored, collateral is secured by DLC. Alice and L15 consensus validate

alternate solution at the pre-production phase are negligible.

[15]While keeping close to the description of DLC by the inventors (Dryja, 2017), we are using the concept and the term in a non-standard way on a few occasions.

[16]CET is Contract Execution Transaction as described in the DLC foundational paper (Dryja, 2017).

[17]Liquidation: Termination of loan contract due to loan collateralization dropping below the minimum for the contract.

[18]Flash Payoff: Termination of a collateralized loan by the user, or upon contract expiration.

the DLC during the contract's creation. Up to this point, all parties have no variability or equivocation.

The contract safely stores Alice's collateral until the conditions for release are satisfied. There are three types of release conditions.

6.3 Loan Payoff

L15 uses the VSD / VC-P for this cross-chain sequence.

Each Payoff CET (fig. 4, 6) is active for a particular time, corresponding to a specific payoff amount that includes changing interest. The number of CETs depends on how frequently the interest is recalculated.

Let's review the payoff sequence associated with an individual CET_i.

To release her collateral, Alice must burn L15$ (fig. 6); burn is the $LEvent_r$. Consequently, $PTrx_r$ is the Bitcoin transaction of collateral payout to Alice: the appearance of a Bitcoin UTXO, unconditionally spendable by Alice's private key, given that the corresponding public key was used at contract initiation. This is considered the end of the release process.

L15 Chain: $LEvent_r$

Burn Request Opcode L15 introduces a new, L15-specific script opcode `OP_BURN_REQUEST` (fig. 6, 3). Among other things, the new opcode signals miners to enforce an L15 consensus rule that enforces the atomicity of the process in the opposite direction: from Bitcoin to L15. Burn is not considered valid unless there is a corresponding release transaction in Bitcoin, whether finalized or in progress. Otherwise, L15 miners will allow Alice to spend her L15$ that she had previously submitted for the burn. Alice's burn request must also contain $sig^{BTCtx}(sk_A)$ – Alice's signature for the Bitcoin chain transaction that pays out her collateral.

Step 1 The initiative for the loan payoff comes from Alice: she creates a burn transaction request via `OP_BURN_REQUEST`.

Step 2 L15vroutenode signs Alice's burn request. Before signing, L15vroutenode validates that Alice's collateral exists and that her loan payoff is of the correct amount. Validation by L15vroutenode is similar to how Bitcoin full nodes validate transactions in the mempool before the miners. Once Alice's burn request passes the validation, L15vroutenode signs it, thus indicating that they have accepted the burn (fig. 6, 2). Signatures of the L15vroutenode members establish a claim to their fees; at the same time,

L15vroutenode reveals the secret $sig_S(q_i)$ (fig. 6, 5), formed like an oracle's broadcast for DLC and used as the second private key for the multisig that locks Alice's collateral.

A corresponding public key was already created at contract negotiation:

$PK_{q_i} = R_S - h(q_i, PK_S, R_S) \cdot PK_S$

$sig_S(q_i) = r_S - h(q_i, PK_S, R_S) \cdot sk_S$

where q_i is the burn ID that corresponds to CET_i:

$q_i = h(amount_i || R_S)$;

$amount_i$ is the amount being burned, which is, following an earlier Oracle metaphor, an analog of the broadcast data,

R_S is the ephemeral public key from L15vroutenode generated for CET_i.

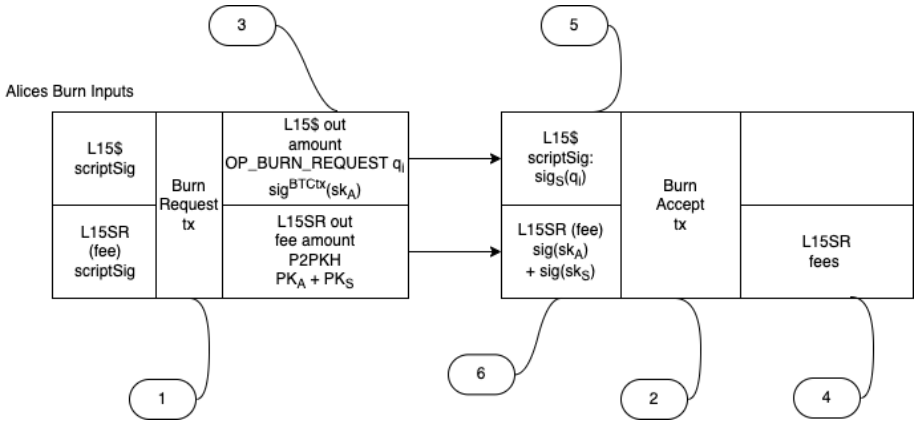

Figure 6: Burning L15$

Note that all the communications between Alice and L15vroutenode happen through the L15 blockchain, i.e., transactions are validated in real-time and mined by the L15 miners.

Bitcoin Chain: $PTrx_r$

Completion of $LEvent_r$ creates the records of all the data necessary for the next Publishing Miner to create a spending transaction for the Validating Miner(s), the first step necessary for $PTrx_r$. Until the next Publishing Miner appears, $PTrx_r$ exists in that form, waiting. Once the Publishing Miner takes it to the next step, the process follows the exact protocol outlined in the VSD / VC-P section.

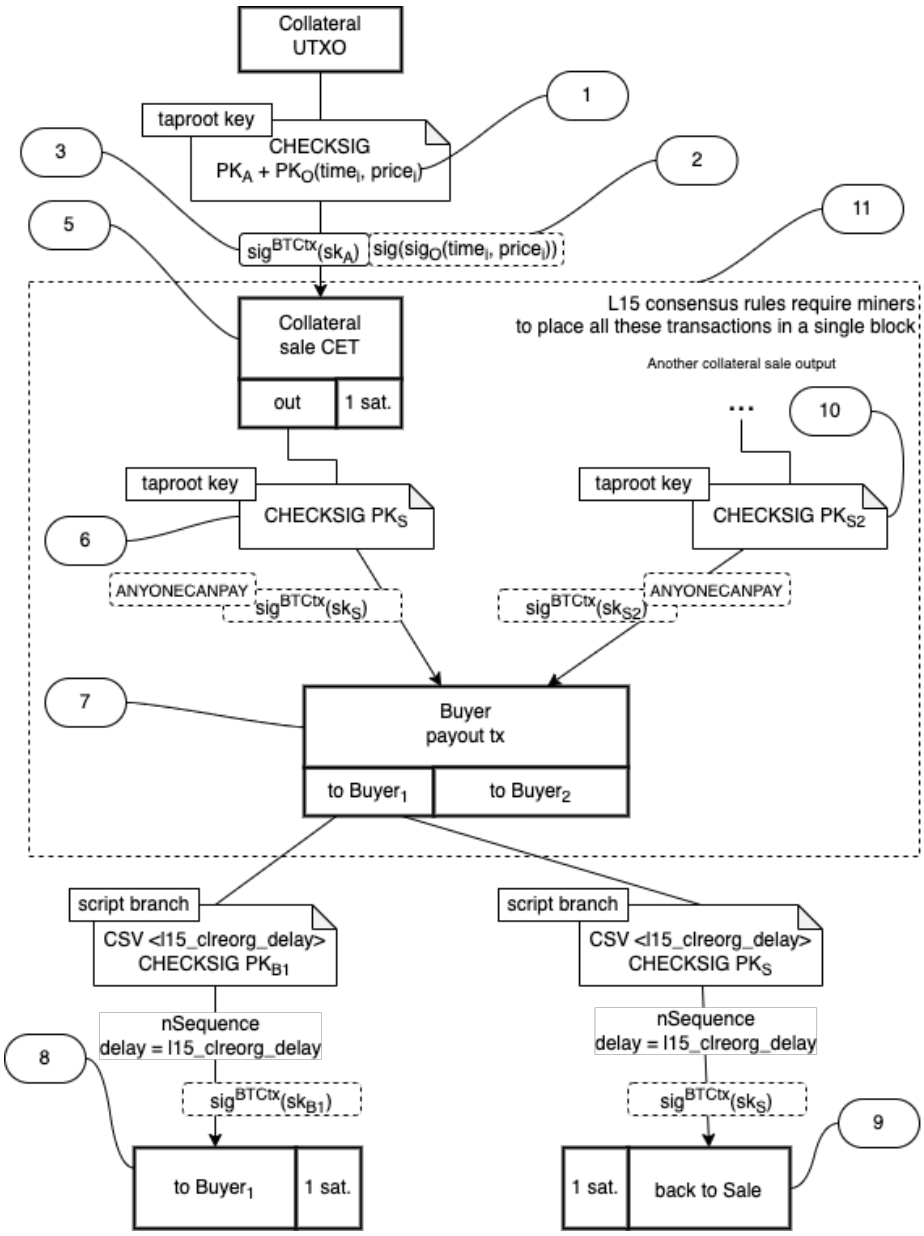

Figure 7: Collateral Sale Transaction

6.4 Liquidation

Liquidation (fig. 4, 7) is triggered trustlessly and executed via the standard means of DLC (Dryja, 2017), using Olivia's broadcasts. Upon the actualization of appropriate conditions, collateral becomes spendable by a CET that enables the L15vroutenode to send it through a process similar to Loan Payoff. Still, the collateral buyers are the ones burning L15$ instead of Alice.

In addition to validating the burn, Validating Miners enforce the consensus rule that collateral goes to the highest bidder (fig. 4, 9).

When the sale is triggered, CET becomes spendable by PK_s(fig. 7, 6), i.e., collateral now depends on the signature of the L15vroutenode. In order to use PoW validation, CET(fig. 7, 5) and spending scripts (fig. 7, 8 and 9) have a one-satoshi output. There is also a consensus rule that all transactions within the dashed box (fig. 7, 11) must be validated and included in the same block by the same Publishing Miner. At that point, payment to the buyer is waiting for $l15\_clreorg\_delay$ ($l15\_clreorg\_delay + to\_self\_delay = T1$; $l15\_clreorg\_delay = T2$) and for the Validating Miner to include one of the possible branches of the transaction (fig. 7, 8 or fig. 7, 9). This is dependent on the Validating Miner spotting any reorgs in the L15 chain which may have resulted in the rollback of buyers' burn of L15$.

6.5 Flash Payoff

The sale transaction (fig. 7) is the same for both Liquidation and Flash Payoff. Note that the CET (fig. 7, 5 and fig. 4, 8) is one of the possible outcomes of CET sets from either Liquidation (fig. 5, 5) or Flash Payoff (fig. 5, 6). In the latter case, the Oracle key check (fig. 7, 1 and 2) does not exist in the script, Olivia's signature (fig. 7, 2) is not supplied. Instead, Alice's signature (fig. 7, 3) becomes a trigger for the sale.

7 Conclusion

We have described the L15 *Trust Model* by presenting data to assist predictions about L15 behaving as expected. The goal was to measure the degree of trust assurance using Bitcoin security as a frame of reference. To summarize, L15 is secured differently in three different functional areas:

1. The L15 chain itself, being a PoW chain, could hypothetically become as secure as Bitcoin, provided that 100 percent of Bitcoin miners merge mine L15, and the rewards are on-par.
2. Contracts in part are Bitcoin scripts, using DLC and other native crypto-

graphic constructs, so these parts can already be considered as secure as Bitcoin, using the same constructs, and the same Oracles in the case of DLC. Threshold multisig stands out here as being novel, but we expect that Bitcoin applications will start utilizing it even before the L15 launch, so our earlier statement holds.

3. The most interesting case is the bridge between Bitcoin and L15, and its core part - the VSD / VC-P method.

   VSD / VCp is used for three types of collateral release: payoff, flash payoff, and liquidation. It comes into play to protect the system from Alice staging a fake payoff and to make sure that in the cases of liquidation and flash payoff Alice's collateral is indeed sold to the highest bidder.

   Admittedly, Bitcoin has an advantage over VSD / VC-P even in the best-case scenario of equal hash powers. One can create an L15-invalid transaction that will be accepted by Bitcoin as valid, i.e. Bitcoin side will not secure it to L15 rules. There is a combination of safeguards in L15 to mitigate this:

   - VSD / VCp creates a way to validate a cross-chain transaction by $1+N$ consecutive *L15minerB*s, which is equivalent to N Bitcoin block confirmations.
   - If the attack through miners is maintained beyond N confirmations, there is a $Majordomo$ to catch a fault. Miners may be interested in stealing a part of the collateral and splitting it with Alice, while for *Majordomo* it is not feasible to break only one contract. Since his wealth depends on the system as a whole, only stealing all of the collateral in the system may be worth the effort.
   - Similarly, L15vroutenode provides an additional pre-transaction safeguard, and they have their bond in Bitcoin to lose.

     Two latter points will not be as assuring if used as standalone, but instead, they compliment PoW validators as a functional extension: L15Vroutenode provides a function of memory, and Majordomo provides an "undo button". Working together, they provide a non-linear positive compounding effect.
   - Also, it is important to keep in mind that each collateral must be attacked separately. Additionally, in case of a fake payoff, a single contract bounty has to be split between Alice, miners, Majordomo, and signers, narrowing the profitability of the attack to a very limited set of circumstances. While the cases of liquidation and flash payoff

exclude Alice from the attack, they involve on average more than a 50% lesser bounty, which evens out the profitability playfield.

The L15 creators make the claim that these safeguards, used together with PoW consensus and Bitcoin-secured elements, form a system capable of assuring trust when left to work unattended as “unstoppable code”.

References

Back, A., Corallo, M., Dashjr, L., Friedenbach, M., Maxwell, G., Miller, A., Poelstra, A., Timón, J., & Wuille, P. (2014, October 22). *Enabling blockchain innovations with pegged sidechains* [White Paper]. https://blockstream.com/sidechains.pdf

BOLT v3, bitcoin transaction and script formats. (n.d.). [Standard]. https://github.com/lightningnetwork/lightning-rfc/blob/master/03-transactions.md https://github.com/lightningnetwork/lightning-rfc/blob/master/03-transactions.md

Cohen, N. (2021, May 07). *Multiple oracle support* [GitHub]. https://github.com/discreetlogcontracts/dlcspecs/blob/master/MultiOracle.md

Drijvers, M., Edalatnejad, K., Ford, B., Kiltz, E., Loss, J., Neven, G., & Stepanovs, I. (2018). *On the security of two-round multi-signatures* [White Paper]. https://eprint.iacr.org/2018/417.pdf

Dryja, T. (2017). *Discreet log contracts* [White Paper]. https://adiabat.github.io/dlc.pdf

Guillyr, T. L., Benthecarman, & Cohen, N. (2020, November 03). *Oracle specifications* [GitHub]. https://github.com/discreetlogcontracts/dlcspecs/blob/master/Oracle.md

ITU. (March, 2000). *International standard 9594-8 ITU-t recommendation x.509 information technology – open systems interconnection – the directory: Public-key and attribute certificate frameworks* [Standard]. https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-X.509-200003-S!!PDF-E&type=items

Komlo, C., & Goldberg, I. (2020, December 22). *FROST: Flexible round-optimized schnorr threshold signatures* [Report]. https://eprint.iacr.org/2020/852.pdf

Poelstra, A. (2015, March 22). *On stake and consensus* [White Paper]. https://nakamotoinstitute.org/static/docs/on-stake-and-consensus.pdf

Trust modeling for security architecture development. (Apr 18, 2003). [Article]. https://www.informit.com/articles/printerfriendly/31546