Civ Kit: A Peer-to-Peer Electronic Market System

Nicholas Gregory, Ray Youssef and Antoine Riard
a@hachiko.dev

Abstract. The Bitcoin protocol enables a global store of value system without a single trusted third party issuing or holding the funds. Participants are only required to have access to a standard computer using a broadband connection. As a second layer, the Lightning protocol enables fast, instant, and cheap value transfers between all participants of the Bitcoin system by rolling out a decentralized network of payment channels. While those protocols enable global value transfers, they do not allow global trading with the same properties.

We propose a new peer-to-peer electronic market system, which enables censorship-resistant and permissionless trading between users of the global Bitcoin system. This design builds on top of the new Nostr protocol for its peer-to-peer order book and relies on the Bitcoin blockchain as a source of truth for its Web-of-Stakes market ranking paradigm. Market trades are locked under Bitcoin contracts to avoid reliance on trusted third parties for dispute arbitration. All market nodes are incentivized by privacy-preserving service credentials backed by Bitcoin payments. This market system should enable global trade of any kind of item all over the world: fiat currencies, goods, services.

License. This work is released into the public domain.

1 Introduction

Global trade has come to rely on a market system characterized by high barriers to entry and dispute mediation costs. Although this system functions for the historical countries that initiated its design, it excludes large segments of the world population, such as the Global South and some Western minorities (immigrants, youth). The architecture of the current market system creates and maintains significant structural asymmetries between population distribution and wealth distribution. Within this system, centralized parties with

low visibility and predictability determine the price and circulation of fiat currencies. This uncertainty generates additional transactional costs for people who must acquire these centralized fiat currencies to participate in global trade. Participants trading across countries often lack standardized and compatible social identification techniques to address their institutional confidence needs. A more decentralized market infrastructure can be desirable with the same properties Bitcoin has achieved as a store of value: censorship-resistance, neutrality, openness.

A peer-to-peer electronic market system is needed to extend the global flows of Bitcoin liquidity, fiat currencies, and goods to any community in a permissionless manner. In this paper, we propose such a system in which two parties can discover trade offers across a set of distributed bulletin boards, find each other, and execute a trade for anything using Bitcoin or Lightning as a clearing layer. No custodians are needed to escrow the coins, as they remain locked under Bitcoin Script trust-minimized escrow contracts. The escrow mechanism can be refined over time to accommodate all types of transactions, from Bitcoin to fiat money to goods and services, by leveraging oracles. Anyone can start a market bulletin board or oracle for anything, and anyone can engage in trading. All the frictions and blockades that have hindered free trade will be reduced to pure technical capabilities.

2 Design Rational

Peer-to-peer electronic markets must possess specific attributes to accomplish the primary goal of offering affordable, censorship-resistant trading on a global scale. A decentralized system comprises a network of peer-to-peer servers that coordinate their operations without depending on trusted third parties [1]. Operating such a server should require minimal computational resources and have low barriers, akin to running a Bitcoin full node.

This market system should be capable of scaling to billions of trades per second with minimal processing bottlenecks while maintaining reasonable bandwidth demands for mobile trading clients. Trade flows should be consistent for all involved participants of a marketplace, meaning that there should be minimal information asymmetries resulting from the market infrastructure design or participants processing capabilities [2].

Market system servers, also known as "functionaries," should be incentivized to enforce accurate operations and provide their services transparently. This "infrastructure-as-a-market" approach should resemble the dynamic membership of mining nodes on the Bitcoin base layer or routing hops on the Lightning Network. The incentive framework should be robust enough to safeguard public servers from client spam.

Furthermore, the market system should not discriminate against participants based on their identity, types of trades, or trade flows. A high level of confidentiality supports this property while also reducing market asymmetries. To harden against censorship attacks, separation of concerns should be

followed. Namely, the market system components have well-defined services scopes and be able to be run by different entities in a interoperable manner.

Although it might be tempting to rely on a global consensus system like a blockchain to establish a decentralized order book of trades, this approach encounters several challenges in practice:

- High level of asymmetries due to latency in convergence mechanisms;
- Low trade throughput as a result of the need for decentralized global consensus;
- High level of trade noise caused by insufficient market specialization unnecessarily consuming bandwidth;
- Restricted flexibility of block content format coming with high deployment cost for new types of market orders ;

3 Background

Bitcoin is a peer-to-peer electronic cash system relying on a large network of independent validating nodes exchanging blocks of transactions [3]. Blocks are chained by a proof-of-work generated in an open fashion by the miner nodes.

3.1 Bitcoin Script

Bitcoin units are represented by a collection of the unspent transaction outputs (the *UTXOs*), that result from the validation of a chain of blocks. These outputs indicate an amount locked by a scripting language known as *Bitcoin Script* [4]. This scripting language allows encoding various *cryptographic puzzles*, where submitting a valid solution authorizes a coin movement. While the most commonly used puzzle involves generating a signature against a pre-committed public key, other available primitives include timelocks, basic mathematical operations, and SHA256 hashing.

These puzzles can be utilize to create *Bitcoin contracts* within a single transaction or a chain of transactions. Practical Bitcoin contracts include payment channels [5], atomic swaps [6], and multi-signature escrows [7]. Proposals to extend the basic set of scripting primitives aim to expand the range of Bitcoin contracts that participants can engage in on the Bitcoin blockchain [8].

3.2 Lightning Network

Scalability of the payment throughput has been a fundamental issue of the Bitcoin system as a peer-to-peer electronic cash proposal. Indeed, the reliance on a chain of blocks being validated by all the nodes of the network, while removing the trust risk of a few points of failure, comes up with the downside of a restraint of the block size and issuance rate to keep validation affordable to hobbyist resources. Bounded block sizes induces a constraint on the payment throughput, and as such on the economic openness of Bitcoin as a day-to-day cash system.

As a palliative solution, second-layers made of a network of Bitcoin contracts have been proposed [9]. Those contracts are *2-parties payment channels*, where the payment throughput between the participants stays confidential. Payment channels states are designed to leak on the blockchain only in case of disagreement between the channel participants. In that situation, the blockchain and its scripting language are adjudicating funds in a trustless fashion based on the submitted pre-signed transactions.

The most known network of payment channels is the Lightning Network [10]. This system enables payments across circuits of channels by chaining off-chain secondary Bitcoin contracts: *Hash-Timelocked Contracts* (HTLC). As of April 2023, the Lightning Network has 16,369 nodes and 74,377 public channels for 5,414 coins locked.

3.3 Onion Messages

In theory, the Lightning Network is designed to ensure a high level of anonymity and confidentiality for payment flows, thereby protecting its users financial privacy. To achieve this goal, the chain of HTLCs employs *onion routing*, meaning that the HTLC message is wrapped in multiple layers of encryption, one for each hop involved in the communication channel [11]. In the most privacy-preserving implementation, this onion routing is constructed by the payer, and intermediate hops do not gain knowledge of the source and destination of the HTLC chain.

Additional techniques have been proposed to enhance Lightning onion messages. The *trampoline* technique allows the delegation of segments of the onion routing path to intermediate nodes [12]. With *route blinding*, the Lightning node pubkey of the payee can be hidden from the payer. This method involves adding a blinding of the final hops of the HTLC chain and an entry point for the payer to use when generating their side of the onion route [13]. The Lightning onion format has been generalized to support the transport of arbitrary messages, beyond just HTLCs [14].

3.4 Offers

The Lightning protocol comes with its own payment request protocol, *the invoice format*. This format includes basic fields such as a payment hash to lock the HTLC contract, a human-readable description of the payment purpose, the pubkey of the payee, and expiration time. It is authenticated by the payee signature [15].

However, this invoice format suffers from many issues: wholeness of the signature, lack of per-user binding, lack of field extractions. On top of it, it requires the payee a static endpoint, such as a website for the invoice distribution. Reliance on a website introduces a payment confidentiality breach, as DNS servers have to be involved, and a security dependency on mainstream PKI for website certification.

A new payment request protocol has been designed for Lightning, *the offer format* [16]. This format enables a 3-phase flows where the location of the payee can be anonymized among the onion routing network. The offers can be bounded to a quantity, extended with new fields, partially signed with a payment key dissociated from the Lightning node public key, and they can be represented by a QR-code friendly character set for mobile usage.

3.5 Nostr

While the Lightning Network onion routing enables low-latency anonymous communications, it does not enable multicast message broadcasting, where a message is delivered to an open set of receiver nodes. Recently, the Nostr architecture has been proposed as an open protocol to enable censorship-resistant social network [17].

The architecture relies on *relays servers* receiving and flooding back events between *clients*. The relays do not communicate with each other in the simplest deployment and migration cost between relays is designed to be minimal.

The clients are identified by a public key and the issued events are countersigned. Key management is the responsibility of the clients. A "post" event can include any structured data, while an emphasis on extensibility and backward-compatibility is aimed for. Clients subscribe to relays of their choice to receive "post" events.

As of April 2023, Nostr network has 250,000 daily users, 1339 relays and 27 millions of posts per day.

4 The peer-to-peer orderbook

4.1 Orderbook Design

Nostr as a communication protocol is selected as it presents two valuable advantages for a censorship-resistant and fault-tolerant orderbook. There is a multicast broadcast mechanism where the trade events can be announced with the same "best-efforts" reliability to a group of interested clients. Additionally, credentials are managed by the clients enabling cheap migration between market boards.

The Lightning onion routing infrastructure is leveraged to add a layer of confidentiality in the benefits of trade events and clients. In comparison to other anonymity network, the Lightning channels offers a native protection against paralyzing denial-of-service attacks.

The protocol numbers 4 entities: bulletin board (i.e a Nostr server associated with a Lightning gateway), maker, taker, onion routing hop.

The protocol follows 3 phases:

- *dissemination*: trade orders are routed from the maker to the board;
- *publication*: trade orders are published from the board to its clients;
- *settlement*: trade orders are concluded between the maker and the taker over the Lightning Network;

4.2 Order dissemination

During the order dissemination phase, trade makers send a set of trade orders via Lightning onion communication channels to an unlimited number of market bulletin boards. A trade order is an extended Lightning offer containing additional information about the escrow contract conditions (oracles used, timelocks, abort options, etc). By design, Lightning offers commit to standard maker information, such as currency, goods or service description, sell amount, expiration block height or UNIX epoch, and maximum quantity of products to be sold.

A trade maker selects an entry point in the Lightning onion routing network by sending an onion message to any Lightning node that accepts inbound onion traffic. The onion message commits to a hidden relay path created by the maker and should be transferred hop by hop until it reaches the bulletin board gateway. Once the onion message arrives at the Lightning gateway, it is fully decrypted, and the contained offer is published on the bulletin board.

For instance, Mary, the maker, wants to sell 100,000 nairas at the price of 0.1 BTC. She prefers to make the fiat payment via bank wire transfer and, in case of a trade dispute, involve an escrow for arbitration after 48 hours. As a first step, she generates an offer with that information. She aims to publish her offer on a bulletin board known for aggregating quality naira trades, like Billy's board. She crafts an onion message destined for Billy, forwarding it through two onion routing hops, Alice and Caroll.

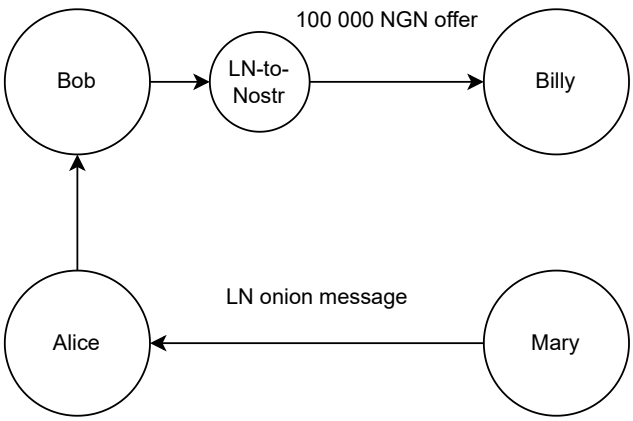

Fig. 1 Order dissemination phase: Mary sends her offer over LN onion routing hops Alice and Bob until reaching Billy's LN-to-Nostr gateway.

4.2.1 Order publication

During the order publication phase, market bulletin boards receive onion messages and distribute the contained trade orders to all their Nostr subscribers.

A market bulletin board consists of a Lightning onion gateway and a Nostr relay, both responsible for basic processing of protocol messages, such as subscriptions, event reception, and event publication. The bulletin board receives onions from its gateway, decrypts the contained offers, and broadcasts them as Nostr events to all its clients who have subscribed to the order trade feed, i.e., the type of product described in the offer.

If a client has been offline at the time of the initial order trade announcement, and the offer has not yet expired, the offer is re-announced as a Nostr event to the offline client.

For instance, when Mary's onion message is received by Billy, it is decrypted one last time and the offer is published on his bulletin board. A Nostr event is then sent to the three bulletin board clients, including Terry.

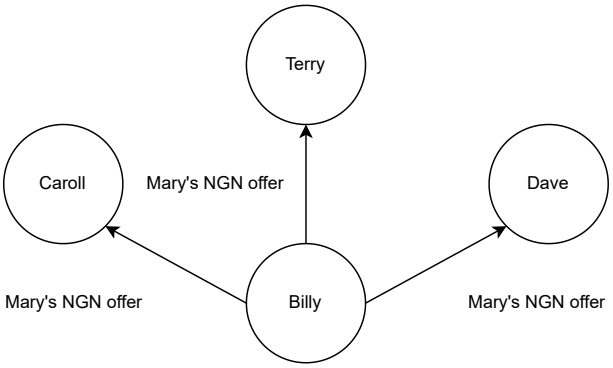

Fig. 2 Order publication phase: Billy relays Mary's offer event to all his clients: Caroll, Terry and Dave.

4.2.2 Order settlement

During the order settlement phase, the takers receive the trade orders as Nostr events, and send a Lightning packet to the makers through the Lightning channels to enter into the trade. Market matching is the responsibility of the takers.

A trade taker connects to an unbounded number of bulletin boards, relaying the takers types of trade of interest. For a type of trade, there can be a disjunction between the set of bulletin boards the maker relays to and the taker subscribes to. However, it is expected takers to connect to as many bulletin boards as they can to gain higher visibility of all the available trade orders.

When a taker receives a trade order, they should parse the escrow contract information, and evaluate if the offers and escrow conditions fulfill their trading requirements/strategy (e.g timelock duration until expiration, number of moderation oracles, etc). In case of success, they should send a corresponding Lightning HTLC from their channels to the maker's blinded path entry point. The Lightning HTLC is modified to support the trade escrow contract.

Once the Lightning HTLC is received by the maker, and assuming the quantity of offered items has not been exhausted by previous trade settlement, the HTLC is accepted and the trade is concluded. Terry cannot backoff from the trade anymore. Further trade operations are being pursued without involvement of the bulletin boards.

For instance, Terry wishes to exchange 0.1 BTC for 100 00 nairas, he receives Mary's matching offer from Billy the bulletin board. Terry forwards a HTLC packet to Mary to enter into the trade through a Lightning payment with blinded path support.

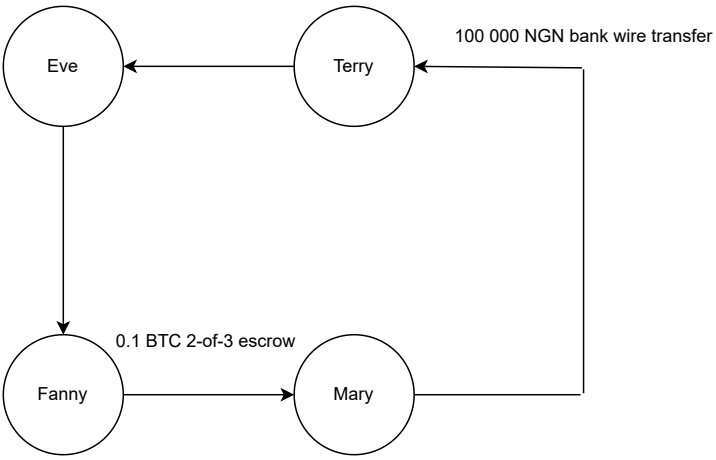

Fig. 3 Order settlement phase: Terry sends an escrowed Lightning payment over LN payment routing hops Eve and Fanny until reaching Mary. Mary sends a bank wire transfer in nairas

5 Orderbook Risks

Orderbook protocol operations are exposed to diverse security risks and attacks:

- sybil attacks;
- onion jamming;
- targeted censorship attack;
- market frontrunning;
- order tampering;
- bulletin board spamming;

There is a risk of a *Sybil attack*, where a counterparty is isolated from the rest of the honest peer-to-peer orderbook and lured into interacting only with spoofed counterparties controlled by an adversary [18]. This risk depends on the peer-to-peer message protocol, network topology, and peering strategy considered. Assuming the orderbook entities rely on DNS seeds to discover the "honest" network and use a flooding mechanism to extend this view, an

adversary could hijack the seeds or exploit implementation weaknesses in the peering strategy.

The dissemination phase relies on Lightning onion communication channels to transport trade orders in a privacy-preserving manner. The current Lightning onion communication channels do not offer a mitigation against *onion jamming*, where an adversary exhausts the allocated bandwidth for onion propagation by Lightning nodes [19]. To address this concern, the Lightning channels topology can be used as a rate-limit mechanism, where a ratio of onion transmission units is allocated for every satoshi allocated between counterparties. Makers with high-volume trade orders will either open more channels or buy out-of-band onion credits. Relying on the Lightning channel topology allows anyone with access to a channel to participate in the peer-to-peer orderbook.

Any maker or taker can be selectively denied access to trade orders by a bulletin board, beyond the restrictions announced by the board's policy (e.g., relaying only specific classes of trades). The risk of *targeted censorship attack* is mitigated by using onions to unlink trade order issuances from their authors, the lack of persistent identity on the offers, and the costless acquisition of a new client identity as a taker (i.e., a new pair of Nostr keys).

Besides censoring counterparties, a bulletin board can censor trades based on their characteristics. As the set of bulletin boards is open, censored types of trades can be relayed by another bulletin board, as long as there is sufficient economic traffic to sustain the operations.

There is a risk of *frontrunning* [20], where the bulletin board withholds a trade order issued by a maker to engage in opportunistic arbitrage as a maker or taker. This frontrunning concern can be reduced by publishing the same trade order on multiple concurrent bulletin boards. If at least one bulletin board is honest, the trade order should be published, and the maker should record the publication timing on all selected boards to monitor them. Unusually slow bulletin boards should be excluded from future publications.

Another variant of frontrunning involves reordering the trade orders relayed to the taker clients, thereby influencing the settlement of the trades. A mitigation strategy is to connect to multiple concurrent bulletin boards, as any reordering should be detected as an anomaly, as long as there is an "honest" board. However, it should be noted that reordering could be the result of the board's policy and, as such, an efficiency improvement in relaying relevant information to its clients.

Another significant risk is order tampering, where the content of the order is altered. The trade order can be optionally signed with semi-persistent keys, which, while introducing a downside in privacy, enforces some integrity for the takers from the bulletin boards. Similarly, the trade order should only be canceled by authenticated requests to prevent a maker from having all its trades canceled by a competitor.

For the bulletin boards themselves, there is a risk of *spam attacks*, where invalid or economically irrelevant orders are massively announced by an adversary. Spoofed orders can be deterred by requesting a Bitcoin payment or committing to collateral toward the bulletin board as part of the incentive framework. The collateral pricing of the trade publication risk can be tailored according to the issuing maker's rank if semi-persistent keys are used.

Lastly, the most efficient bulletin boards concentrating the best orders in their class of trades along time are at risk of becoming systematically important market actors. Their failure or compromise can provoke significant disruptions with lasting effects on the peer-to-peer markets operations. While this risk is hedged by the ability of makers and takers to replicate or duplicate their trading operations on another board at low-cost, additional safety can be introduced by running the bulletin board as a federation [21].

6 The trade escrow Bitcoin contracts

6.1 Types of oracles

In addition to a peer-to-peer order book, functional peer-to-peer electronic markets require various types of oracles:

- *Moderation oracles*: They intervene in contentious trades to adjudicate the funds;
- *Know Your Peer (KYP) oracles*: These attest to social attributes of the trade counterparties or social properties of the trade material (e.g gift card authenticity);
- *Real-world oracles*: They attest to real-world events (e.g shipment delivery).

All types of oracles can participate in trade operations. Both makers and takers can verify their trade counterparties based on the trade material authenticity as attested by KYP oracles (e.g bank information validity). Once the trade is concluded and in case of a dispute, moderators can use the information attested by KYP or real-world oracles to examine the state of the trade flows (e.g a mining ASIC delivered to the payer). This attested information should improve the quality of moderation decisions.

Like the bulletin boards, oracles can be selected openly by trade counterparties. The moderator oracle can be directly integrated into the Bitcoin escrow contract between trade counterparties. The level of integration can vary from a simple setup to advanced ones. Furthermore, the number of oracles per type can range from a single one to an *N-of-M* policy.

6.2 Simple Escrow

Bitcoin Script can be utilized to create simple escrow using multi-signature opcodes. For instance, a straightforward script policy fulfilling the trade requirements of the previous naira-BTC trade example can be as follows:

"If Mary sends the naira bank wire transfer to Terry, Mary can unlock the BTC funds with Terry's cooperation through their joint signatures. After 100 blocks, Mary can unlock the BTC funds with the cooperative signature of Olivia, the moderation oracle. After 100 blocks, Terry can cancel the BTC funds withholding with the collaborative signature of Olivia. After 200 blocks, the BTC funds are returned to Terry."

These Bitcoin escrows can be built on top of Lightning payment channels, provided that a preimage is released by the maker to maintain the operational correctness of the chain of HTLCs across the Lightning payment path. This preimage technique should allow the Lightning routing hops to route escrow contracts without requiring software support for the on-chain or off-chain Script resolution of the escrow paths.

6.3 Advanced Escrow

The activation of the Schnorr/Taproot soft-fork over the Bitcoin blockchain enables the support of point-time-locked contracts over Lightning channels [22]. Namely, the fundamental idea is to replace the hash-lock mechanism by the reveal of a Schnorr signature satisfying a public key.

Payment points themselves allow the introduction of secret sharing for general access structures as a building block for more advanced Bitcoin escrow contracts [23]. Formally, a secret sharing for general access structure is a technique to share a secret K (e.g a Lightning payment point) in such a way that a *N-of-M* combination of partial secrets allows the reveal of the complete secret K itself.

The *N-of-M* combination could be any logical circuit of oracles (price, moderators, "know your peer") following the trade conditions negotiated by the counterparties. To reveal their partial secrets, the oracles can request the satisfaction of their own out-of-band policy. E.g, a moderation oracle can enforce that trade proofs should be in conformity with the moderation rules and should be communicated in time-sensitive fashion to perform adequate adjudication.

Assuming the usage of Taproot, those oracle circuits could be gated under time-locked Taproot branches [24]. The key-path spend could always be used in case of agreement on the trade between counterparties. Iterations of Bitcoin tooling such as Miniscript should allow the trade counterparties to generate and commit such advanced Script policy [25].

Being out-of-band, the oracle policy rules can leverage advanced non-standard Bitcoin cryptosystems, such as decentralized identifiers or homomorphic commitments on structured messages. Decentralized identifiers are a new type of cryptographic identifier enabling verifiable and decentralized digital identity [26]. They can point indifferently to a person, an organization or a real-world object. Homomorphic commitments enable partial reveal of message properties such as an amount or a timestamp.

For instance, Mary and Terry have a dispute on their 0.01 BTC for 100 000 nairas trade. Their Bitcoin escrow contract includes a Taproot branch gated under an absolute timelock of 100 blocks. After 100 blocks, if Ketan the KYP

oracle attests the authenticity of Mary's bank and Olivia and Olive attest the bank wire transfer receipt authenticity, the funds are unlocked to Mary.

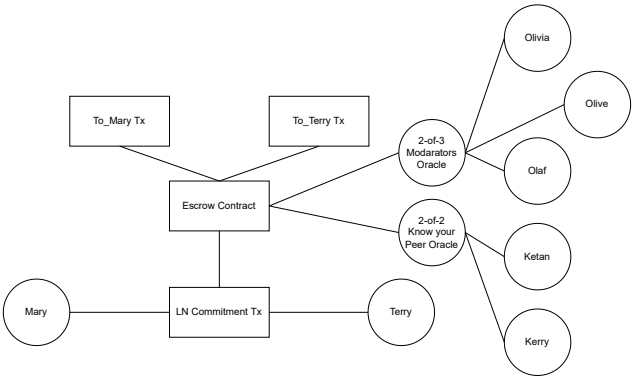

Fig. 4 Advanced Escrow: Mary and Terry have a pending trade escrow open on their LN commitment transaction. The escrow can be settled by Mary or Terry signature and a valid combination of Olivia, Olaf and Olive as moderation oracles and Ketan and Kerry as Know your Peer oracles

7 Order in the market: the Web-of-Stakes

7.1 Spamming issues in peer-to-peer electronic markets

While a peer-to-peer orderbook and Bitcoin escrow contracts establish the foundation for a peer-to-peer electronic market, it is essential to manage the high volume of information to prevent spam from paralyzing operations.

Spam deterrence has long been a challenge in the open Internet, where free public services are constantly at risk of denial-of-service attacks by careless or malicious users [27]. Authentication based on passwords and PKI has been one way to bridge trust gaps in Internet security architecture, while other solutions like proof-of-work have also been explored [28].

In peer-to-peer electronic markets, spamming issues are multifaceted. Bulletin boards face trade order spam, where the order format is either invalid or designed for denial-of-service, or even more severely, trade order flooding, where orders are economically irrelevant (e.g., trade maturity too far in the future, swaps to currencies with no demand, exorbitant markup fees, etc.). Even though some orders' lack of relevance can be identified based on apparent anomalies, the relevance of market information remains a dynamic qualification subject to inherent market forces.

For bulletin boards, the challenge is to filter out trade order spam without hindering spontaneous market forces. Drawing this boundary should be done automatically, without introducing vectors for censorship.

On the other hand, trade takers face the issue of lazy or malicious counterparties who refuse to honor their issued trade orders by not entering into

Bitcoin escrow contracts. Furthermore, trade participants may exhibit a low level of cooperation by not settling operations off-chain, thus burdening their counterparties with on-chain fees.

An ideal ranking system for peer-to-peer electronic markets should allow for sorting both the quality of trade orders and counterparties. Involving a centralized authority that captures market participant characteristics and weighs them based on a custom algorithm would introduce a trusted third party in the market operations. Web-of-trust, while peer-to-peer in its function, does not scale well for large-scale abstract economic interactions, where direct or indirect social connections between market participants cannot be assumed [29].

7.2 The Web-of-Stakes

The Bitcoin blockchain offers a source of economic relevance tied to pseudonymous identities: the UTXO set. Assuming a zero-knowledge proof system with support for arbitrary computations, attributes of UTXO can be asserted in privacy-preserving fashion (satoshis amount, witnessScript, UTXO age). If the Bitcoin Script can be inspected, a third-party can verify the funds are locked under a valid LN channel funding script and cross-check if it has been announced as a public channel over LN gossips.

All the stakes certificates (i.e a privacy-preserving proof-of-UTXO ownership) for a Lightning node can be collected and their private keys counter-sign a "stake public key" [30]. Akin to PGP, this public key represents a Lightning node economic weight in the channel topology. Under the observation that along time Lightning liquidity should be allocated efficiently, this economic weight assumption should hold.

This economic weight is dynamic in function of your channel opening and closure, and a third-party should prune out the stakes certificates for the closed channels from your economic weight.

This "stake public key" can be used to sign a maker offer. This can be leveraged by the board to compute a ratio between your economic weight and all the historical offers under the same public key to estimate your market-making performance. If the offers are counter-signed by the board and timestamped in the chain, a global historical orderbook can be built and as such a global market-making score for the "stake public key" can be generated.

Assuming cooperation of your Lightning channels, zero-knowledge proofs for second-stage channel escrow transactions can be issued, therefore attesting to the contract flows during a time period. Those flows attestations can be leveraged to refine the economic weight attached to a "stake public key".

Ranking algorithms can assign "stake public key" in range in function of the number of stakes certificates and velocity trades e.g Tier 1, Tier 2 and Tier 3. As a "stake public key" is counter-signed by stakes certificates shared with high-tier entities, its score should increase.

This scheme unlinks the trades from the UTXOs themselves. Additionally, it should be robusts against spammy adversaries, as building a sane economic

weight consumes multi-dimensional resources: Bitcoin capital, chain blocks, channels topology, history of trade settlement. The base decision factors (channel UTXOs, timestamped offers, off-chain packets, network topology) enable to build specialized ranking algorithms. Participant can arbiter between the level of information they wish to attach to their "stake public key" and the confidentiality of their economic flows.

This "Web-of-Stakes" system and associated economic ranking is trustless, meaning there is no trusted external party required for correct operation, apart from the Bitcoin blockchain. Stake certificates can be obtained from market participants or rank proof servers. With only knowledge of the UTXO set, a market user can build a local view of the economic relevance of its counterparties and the associated trades without introducing persistent identities. This ranking system should lead to the emergence of a decentralized and robust web of confidence between market participants.

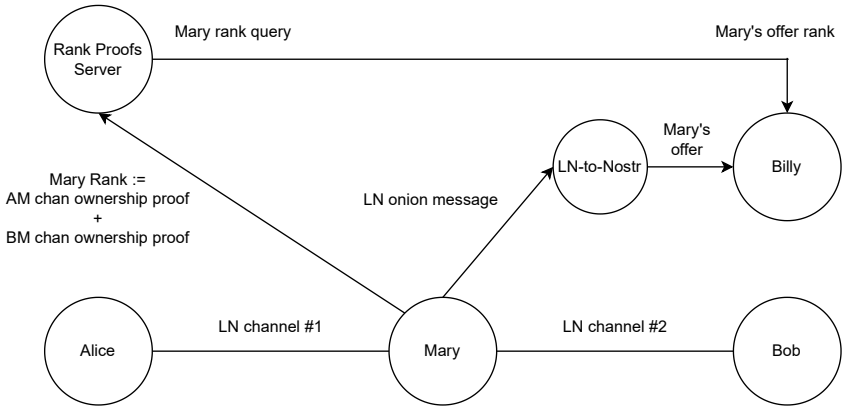

Fig. 5 Web-of-Stakes rank query: Mary owning 2 Lightning channels with Alice and Bob submits proofs to the rank proofs server. Later on, when Mary sends an offer to Billy, she attaches a signature-of-stakes. Billy can leverages this signature to obtain Mary rank from the server and decide or not to publish the offer

8 The incentives of market functionaries

This peer-to-peer electronic market design adheres to an "infrastructure-as-a-market" paradigm, where all services are competitively provided by functionaries, similar to mining nodes on the base layer or routing hops on the Lightning Network. There is no formal barrier to entry for new service providers, nor any "single point of failure" or privileged parties coordinating the network.

This openness applies to all types of market functionaries that make up the system:

- *Bulletin boards*
- *Rank proof servers*

– *All types of oracles* (moderators, "know your peers")

These functionaries can be Nostr relays with protocol extensions dedicated to their market services. Market services and their associated policies can be initiated by client software vendors, announced over the Lightning Network gossip network, or promoted on bulletin boards themselves as new Nostr kinds.

Market functionaries are named as such because their correct operations are fully deterministic based on the received events and announced policies. Encouraging operational correctness should be rewarded with Bitcoin payments or equivalent monetary compensation. Deviations from correctness should be penalized by client ranking algorithms, akin to the scoring of routing hops over the Lightning Network.

Over time, and assuming reliable ranking algorithms, the most efficient, inventory-rich, and available market functionaries should thrive. While basic market services are expected to be homogeneous (e.g., a user can access trade orders for goods A from any bulletin board supporting inventory A), a heterogeneity in quality is anticipated. The membership of market functionaries is dynamic, allowing any new functionary to enter the competition and, if efficient, establish itself among the top market functionaries.

These Bitcoin payments can be intermediated by introducing privacy-preserving credentials similar to the Privacy Pass architecture for user authentication towards HTTP servers [31]. These credentials should maintain user confidentiality by unlinking service rights purchases from their consumption. This unlinking prevents selective denial of service fulfillment requests from users based on payment metadata. Additionally, these credentials should enable rewards for well-behaving users, fine-grained resource control management, and dynamic repricing of market services [32].

Future extensions of the incentives framework should enable flexible and trust-minimized revenue sharing between the market functionaries and the makers or takers, as such aligning incentives.

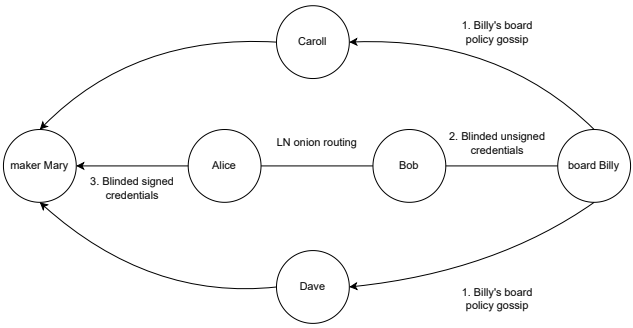

Fig. 6 Credentials issuance phase: Billy the bulletin board announces its order publication policy and the associated fees through the LN gossips. Mary the maker discovers the gossip, send blinded unsigned credentials with proof-of-payment and then receives from Billy the credentials signed.

For instance, Billy the bulletin board can establish a service policy for the next 3 months period requesting that all offers should be attached with credentials worth 1000 sats. Those credentials would allow the offers to stick on Billy's board for 1 month. If the offers publishers have low ranking scores, Billy could request an additional 1000 sats. If Billy's board is well-ranked among all the bulletin boards, a "promotion" fee could be requested to prioritize the offers processing.

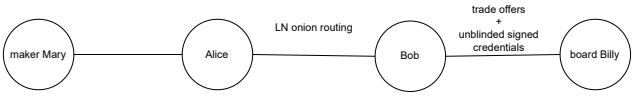

Fig. 7 Credentials redemption phase: Mary the maker attached the credentials to her trade offer and sends her over LN onion routing to Billy. This phase is embedded in the order dissemination phase.

9 Applications

9.1 Bitcoin financial contracts

Peer-to-peer electronic markets are primarily designed for currency trading. However, due to the flexible offers format and the capabilities of Bitcoin Script, this peer-to-peer market infrastructure can support a wide variety of Bitcoin off-chain contracts such as coinjoin, discreet log contracts, hashrate derivatives, and lightning liquidity ads.

All these contracts rely on pre-signed transactions committed by two or more participants. While the contracting mechanisms are trust-minimized, there is no standard decentralized mechanism to match the supply and demand sides of these off-chain contracts.

9.2 Bitcoin Services Providers discovery

One of the design goals of a peer-to-peer electronic market is to establish infrastructure servers as market services, where components can be logically substituted by competing components, at least for their basic functionalities. Any bulletin board or rank proof server can be replaced or used concurrently with another bulletin board in a dynamic fashion.

This discovery mechanism can be extended to existing Bitcoin and Lightning infrastructure providers, such as watchtowers [33], light client servers [34], backup servers, liquidity services providers [35]. This flexible discovery mechanism enables to untangle a client from the default servers seeded by software vendors, thus improving the decentralization of the Bitcoin ecosystem.

9.3 Real-world Goods Market

The peer-to-peer electronic market infrastructure is versatile enough to adapt to real-world goods trading, beyond the exchange of pure commodities. Combined with extended escrow capabilities (timelocks, n-of-m), the escrow script can adapt to the operational constraints of physical goods delivery. For example, a commodity delivery escrow could include a pubkey from every participant in the logistical chain.

Numerous classes of global trade could be traded on the bulletin boards and exchanged under advanced Bitcoin escrow contracts, such as oil and gas, food commodities, mining and AI chips.

10 Conclusion

We have had Bitcoin as a system of peer-to-peer electronic cash for fourteen years, which does not rely on trust. However, Bitcoin cannot reach the masses of global users in daily use unless it can interact with the existing world of fiat currencies, goods, and services trading. We propose a system of peer-to-peer electronic markets without relying on trusted third parties. We start with a peer-to-peer order book that relies on the Nostr client-server architecture and the Lightning onion routing mechanism. To solve trade execution, we depend on Bitcoin escrow contracts, where trade logic can be backed by trust-minimized oracles. Spam deterrence and the ordering of market information are realized through the introduction of the "Web-of-Stakes" paradigm, where the Bitcoin UTXO set and overlay semantics serve as a trustless source of truth. The network nodes fulfilling the system operations constitute a dynamic membership in competition. The correctness of these node operations is incentivized by Bitcoin payments with minimal coordination.

References

[1] Szabo, N.: Trusted Third Parties are Security Holes. https://nakamotoinstitute.org/trusted-third-parties/

[2] Akerlof, G.A.: The market for "lemons": Quality uncertainty and the market mechanism. The Quarterly Journal of Economics (1970)

[3] Nakamoto, S.: Bitcoin: A Peer-to-Peer Electronic Cash System. https://nakamotoinstitute.org/bitcoin/

[4] Script. https://en.bitcoin.it/wiki/Script

[5] Hearn, M.: Anti DoS for tx replacement. https://lists.linuxfoundation.org/pipermail/bitcoin-dev/2013-April/002417.html (2013)

[6] Nolan, T.: Re: Alt chains and atomic transfers. https://bitcointalk.org/index.php?topic=193281.msg2224949msg2224949 (2013)

[7] Andresen, G.: M-of-N Standard Transaction. https://github.com/bitcoin/bips/blob/master/bip-0011.mediawiki (2011)

[8] Optech: Covenants. https://bitcoinops.org/en/topics/covenants/

[9] Todd, P.: Near-zero fee transactions with hub-and-spoke micropayments. https://lists.linuxfoundation.org/pipermail/bitcoin-dev/2014-December/006988.html (2014)

[10] Poon, J., Dryja, T.: The Bitcoin Lightning Network: Scalable Off-chain Instant Payments. https://lightning.network/lightning-network-paper.pdf (2016)

[11] Kate, A., Goldberg, I.: Using sphinx to improve onion routing circuit construction. In: 14th International Conference on Financial Cryptography and Data Security (2010)

[12] Teinturier, B.: Trampoline Routing. https://github.com/lightning/bolts/pull/829 (2020)

[13] Teinturier, B.: Route Blinding. https://github.com/lightning/bolts/pull/765 (2020)

[14] Russell, R.: Onion message support. https://github.com/lightning/bolts/pull/759 (2020)

[15] Invoice Protocol for Lightning Payments. https://github.com/lightning/bolts/blob/master/11-payment-encoding.md (2017)

[16] Russell, R.: Flexible Protocol for Lightning Payments. http://bolt12.org/bolt12.html (2020)

[17] nostr - Notes and Other Stuff Transmitted by Relays. https://github.com/nostr-protocol/nostr (2022)

[18] Douceur, J.R.: The sybil attack. In: Peer-to-peer Systems: First International Workshop (2002)

[19] Teinturier, B.: Onion messages rate-limiting. https://lists.linuxfoundation.org/pipermail/lightning-dev/2022-June/003623.html (2022)

[20] Zhou, L., Xiong, X., Ernstberger, J., Chaliasos, S., Wang, Z., Wang, Y., Qin, K., Wattenhofer, R., Song, D., Gervais, A.: SoK: Decentralized Finance (DeFi) Attacks. Cryptology ePrint Archive, Paper 2022/1773. https://eprint.iacr.org/2022/1773 (2022). https://eprint.iacr.org/2022/1773

[21] Optech: Fedimint. https://fedimint.org (2023)

[22] Optech: Point Time Locked Contracts. https://bitcoinops.org/en/topics/ptlc/

[23] Kohen, N.: A Payment Point Feature Family (Multisig, DLC, Escrow, ...). https://lists.linuxfoundation.org/pipermail/lightning-dev/2019-October/002213.html (2019)

[24] Wuille, P.: Taproot: SegWit version 1 spending rules. https://github.com/bitcoin/bips/blob/master/bip-0341.mediawiki (2020)

[25] Wuille, P.: Miniscript. https://bitcoin.sipa.be/miniscript/ (2019)

[26] W3C: Decentralized Identifiers (DIDs) v1.0. https://www.w3.org/TR/2022/REC-did-core-20220719/ (2022)

[27] Postel, J.: On the Junk Mail Problem. https://www.rfc-editor.org/rfc/rfc706.html (1975)

[28] Back, A.: Hashcash - A Denial of Service Counter-Measure. http://www.hashcash.org/papers/hashcash.pdf (2002)

[29] Wikipedia: Web of trust. https://en.wikipedia.org/wiki/Web$_o f_t rust$(2023)

[30] Naumenko, G., Riard, A.: Mitigating Channel Jamming with Stakes Certificates. https://lists.linuxfoundation.org/pipermail/lightning-dev/2020-November/002884.html (2020)

[31] Davidson, A., Goldberg, I., Sullivan, N., Tankersley, G., Valsorda, F.: Privacy pass: Bypassing internet challenges anonymously. In: Proceedings on Privacy Enhancing Technologies (2018)

[32] Optech: Staking Credentials. https://bitcoinops.org/en/newsletters/2022/11/30/repu credentials-proposal-to-mitigate-ln-jamming-attacks (2022)

[33] Dryja, T.: Unlinkable Outsourced Channel Monitoring. https://btctranscripts.com/scalingbitcoin/milan-2016/unlinkable-outsourced-channel-monitoring/ (2016)

[34] Hearn, M.: Understanding the bitcoinj security model.

https://bitcoinj.org/security-model (2013)

[35] Sheinfeld, R.: Introducing Lightning Service Providers. https://medium.com/breez-technology/introducing-lightning-service-providers-fe9fb1665d5f (2019)