GoMining BTC Instant Payment Protocol

White Paper

Dmitrii Ivanov
GoMining

Version 1.3.0 — April 25, 2026

Abstract

This White Paper presents the architecture of the GoMining BTC Instant Payment Protocol, a Bitcoin payment system enabling instant point-of-sale transactions, no direct Bitcoin network fee charged to users, and publicly verifiable user holdings in on-chain multisig UTXOs without wrapped assets, sidechains, or synthetic BTC representations.

The system is built around a 2-of-3 multisig wallet model with conditional recovery and pre-authorized transactions that enable instant operator-level payment acceptance at the point of sale, distinct from Bitcoin network confirmation. Every instant payment is a real Bitcoin transaction signed by the user's key over their multisig UTXO. While held at the user's 2-of-3 multisig address, funds remain under the user's cryptographic co-control: GoMining alone cannot spend them. Instant payments do not transfer funds to GoMining. At on-chain batch settlement, the latest user-authorized cumulative amount is transferred to a GoMining-controlled output, from which merchant payouts are reconciled as commercial settlement; between on-chain settlement and commercial merchant payout, the settled amount is held under operator custody and merchants are unsecured operator creditors for that amount. Any residual UTXO value above the cumulative amount is returned to the user's multisig as change. GoMining covers network fees and reduces the effective on-chain fee per payment through batching.

The protocol requires no changes to Bitcoin itself and is designed for seamless integration into existing POS infrastructure through SDK-based deployment. It is also built as a wallet integration ecosystem: GoMining provides a reference wallet implementation, and Bitcoin-capable third-party wallets can offer instant-payment functionality to their users through partnership with GoMining and adoption of the wallet integration SDK (Section 10). The result is a scalable payment layer combining instant operator-level payment acceptance with low-cost on-chain batch settlement and user self-custody of uncommitted balances.

Contents

1 Introduction

1.1 Motivation

Bitcoin is widely adopted as a store of value and a censorship-resistant settlement layer. However, using Bitcoin for day-to-day payments remains difficult. On-chain transactions are slow to confirm, fees can spike unpredictably, and merchants rarely have tooling that fits into familiar point-of-sale (POS) workflows. As a result, most "Bitcoin payments" today are processed either through fully custodial services, where users do not control their keys, or through specialized off-chain protocols that require complex channel management and are not easily integrated into existing merchant infrastructure.

For an ecosystem like GoMining, which already operates with BTC and has a large user base, there is a strong incentive to provide instant BTC payments with a UX comparable to traditional card payments, while preserving as much of Bitcoin's non-custodial and trust-minimized properties as possible.

1.2 Goals of the system

- **Real Bitcoin custody on L1.** User funds are held in native Bitcoin P2WSH multisig outputs on Bitcoin mainnet, without wrapped assets, sidechains, or synthetic BTC representations. The instant authorization layer and the commercial merchant reconciliation layer are anchored to these L1 UTXOs.
- **Instant merchant acceptance.** The GoMining operator returns an *Approved* response to the merchant at the point of sale as soon as the user has signed the updated cumulative pre-authorization. This operator-level acceptance is independent of Bitcoin network confirmation and yields a user experience comparable to traditional card processing.
- **Non-custodial security.**[1] Users retain control of funds via a 2-of-3 multisig architecture in which no single party can spend unilaterally, together with an independent time-delayed recovery path for unilateral user exit.
- **No direct Bitcoin network fee for users.** Users are not charged a Bitcoin network fee on instant payments. GoMining absorbs the on-chain fees required to settle cumulative authorizations (including the CPFP child transaction fees described in Section 6) and recovers these costs through merchant service fees rather than charging users directly.
- **Low effective on-chain cost per instant payment.** Aggregating multiple per-user settlement transactions under a single CPFP child amortizes the network fee cost across the entire batch, lowering the effective on-chain cost per instant payment compared to settling each payment as an individual Bitcoin transaction.
- **Easy integration for merchants.** Deployment is enabled through an SDK-based POS solution compatible with existing terminals and merchant workflows.
- **Wallet integration ecosystem.** Any Bitcoin-capable wallet — multi-chain consumer wallet or Bitcoin-native wallet — can offer instant-payment functionality to its users through

[1] Throughout this paper, *non-custodial* is used in the industry sense applicable to threshold-multisig and MPC wallet architectures: no single party — user, operator, or recovery custodian — can unilaterally spend user funds during normal operation, and the user has a unilateral exit path (see Section 7.2). It does not imply self-signed single-key custody. This property applies to funds held at the 2-of-3 multisig address; see Section 6 for the on-chain settlement phase, during which user-authorized amounts are transferred to an operator-controlled output pending commercial merchant payout.

partnership with GoMining and adoption of the wallet integration SDK. The 2-of-3 multisig is constructed from the user's public key (supplied by the wallet via the SDK), GoMining's co-signing key, and a recovery custody key from a custody provider approved by GoMining. The wallet integrator selects a custodian per integration from the approved set, or, subject to GoMining's approval, may itself serve as the custody provider. GoMining ships a reference wallet; broader wallet partnerships are on the roadmap (Section 10).

1.3 Design principles

To meet these goals, the protocol is built around the following design principles:

- **2-of-3 multisig as the base custody model.** Each user wallet is a 2-of-3 Bitcoin multisig where one key is held by the user, one by GoMining, and a third is a recovery key held offline by a trusted external custody provider and released to the user under the conditions described in Section 7.2 (operator-coordinated proof of settlement or fallback time-based delay).
- **Pre-authorized Bitcoin transactions for instant spending.** Every instant payment is a real Bitcoin transaction: the user signs a PSBT spending their multisig UTXO with their User Key. Users sign a cumulative pre-authorization value that is updated with every instant payment, giving GoMining a cryptographic upper bound on the amount it may settle on-chain for that user, while leaving GoMining unable to spend user funds beyond the latest user-authorized value. On-chain settlement timing is subject to Bitcoin network conditions; merchant payouts are settled by GoMining as part of merchant agreements.
- **Batched on-chain settlement.** Instead of broadcasting every individual POS payment on-chain, GoMining settles each active user's cumulative authorization through its own per-user Bitcoin transaction and aggregates a group of such transactions into a single Child-Pays-For-Parent (CPFP) child transaction that pays the market-rate fee for the entire package. Per-user outputs go to an operator settlement address; merchant reconciliation is handled separately by GoMining as part of its commercial settlement processes and is not represented as individual outputs in the settlement transactions (see Section 6).
- **Managed settlement propagation via Stratum V2.** Per-user settlement transactions and their shared CPFP child transaction are held in an operator-side staging queue after construction. From this queue the parent-plus-child package is submitted, via the Stratum V2 Job Declaration sub-protocol, to GoMining-operated mining infrastructure, which includes it in block templates it produces. At the time of writing, GoMining produces on average approximately two blocks per day; for such blocks, the package is included directly. For blocks produced by other mining pools, the parent transactions and the CPFP child reach miners through standard public-mempool propagation. The CPFP child is always attached (including for blocks produced by the GoMining pool) so that the effective package feerate is market-competitive; this preserves normal mining payouts for third-party miners hashing on the GoMining pool and makes the package attractive for inclusion in blocks produced by other pools. Submission defaults to the Stratum V2 path; at operator discretion — for example during infrastructure constraints or elevated network load — the package may also be broadcast to the public Bitcoin network. Stratum V2 is a publicly specified Bitcoin mining protocol with open-source reference implementations; all settlement transactions remain valid, standard Bitcoin transactions, and the arrangement does not modify Bitcoin consensus or transaction standardness rules.
- **Self-custody of uncommitted balances.** While held at the user's 2-of-3 multisig address, user BTC cannot be moved without the user's signature. Instant payments do not transfer

funds to GoMining, do not lock user liquidity under operator control, and do not require GoMining to hold a spendable balance of user assets. On-chain batch settlement transfers the latest user-authorized cumulative amount to a GoMining-controlled output; from that point the settled amount is held under operator custody until commercial merchant payout, and merchants hold unsecured operator credit for that amount. Any residual UTXO value above the cumulative amount is returned to the user's multisig as change and remains self-custodied.

- **Conditional recovery and monitored exit.** The recovery signing capability is held offline by a trusted external custody provider and is not used during normal operation. It is released to the user under one of two conditions, whichever occurs first: (a) the operator confirms on-chain settlement of any pending instant-payment state for the user, or (b) a configured safety delay T has elapsed since the recovery request — ensuring that operator unavailability cannot indefinitely block recovery. See Section 7 for the full recovery flow.
- **Compatibility with the base Bitcoin layer.** Uncommitted user funds reside on the Bitcoin main chain in standard P2WSH multisig outputs. The system does not require changes to the Bitcoin protocol; the multisig address is recognized by any Bitcoin-aware wallet, node, or block explorer as a standard on-chain address. The instant-payment workflow itself (cumulative pre-authorization, PSBT exchange with the operator, recovery flow) requires a wallet that integrates the protocol via the wallet integration SDK; GoMining ships a reference wallet, and broader wallet partnerships are on the roadmap (Section 10).
- **Merchant-friendly economics.** The effective on-chain fee per instant payment decreases as more per-user settlement transactions are aggregated under a single CPFP child, because the package fee is amortized across the batch. On-chain cost per payment is therefore lower than settling each payment as an individual Bitcoin transaction. Comparisons with channel-based systems such as Lightning differ in kind: Lightning has no per-payment on-chain fee, only channel open/close fees.
- **Compatible with existing retail systems.** No new hardware or protocol-specific network is required; merchants integrate through a standard SDK for Android-based POS terminals.

This paper builds on these principles to describe the full architecture of the GoMining BTC Instant Payment Protocol.

2 System Overview

The GoMining BTC Instant Payment Protocol provides a 2-of-3 multisig wallet architecture and an instant authorization layer built on native Bitcoin multisig outputs. While funds are held at the user's multisig address, they remain under the user's cryptographic co-control and cannot be spent by GoMining alone. Instant payments enable merchant-side acceptance without transferring custody of user funds to the operator, while on-chain batching reduces the effective settlement cost per payment. A key property of the protocol is the separation of two custody phases: while user BTC is held at the 2-of-3 multisig address, GoMining cannot move it unilaterally; on-chain batch settlement then transfers the latest user-authorized cumulative amount to a GoMining-controlled output, at which point the settled amount is held under operator custody pending commercial merchant payout. Residual UTXO value is returned to the user's multisig as change and remains self-custodied.

The system consists of three core components: (1) a 2-of-3 multisig wallet model with a time-delayed recovery key, (2) pre-authorized transactions that allow instant spend authorization,

and (3) a batching mechanism that aggregates per-user cumulative authorizations into a Bitcoin settlement package consisting of per-user transactions and a shared CPFP child.

2.1 2-of-3 multisig model

Each GoMining wallet is implemented as a standard Bitcoin 2-of-3 multisignature address, where:

- one key is controlled by the user (**User Key**),
- one key is controlled by GoMining (**GoMining Key**),
- one key is a time-delayed recovery key (**Recovery Key**) whose signing capability is securely held by a trusted external custody provider and released to the user after a predefined delay without requiring cooperation from GoMining.

Any transaction spending UTXOs at the multisig address requires signatures from at least two of the three keys. In typical operation, payments are signed by the user and GoMining. GoMining cannot spend user funds independently, and users cannot spend funds without either cooperating with GoMining or initiating the recovery process. This preserves non-custodial security while enabling instant authorized payments.

2.2 Key roles

- **User:** holds the User Key, initiates payments, and has the ability to exit to self-custody (User Key + Recovery Key) through the Conditional Recovery flow.
- **GoMining:** co-signs authorized transactions, validates POS payment requests, and performs periodic batched settlements on-chain.
- **Recovery mechanism:** provides unilateral exit for the user with a time delay, ensuring safety against withheld signatures or operational failure.

While held at the multisig address, funds remain on-chain under the 2-of-3 cryptographic spending condition, rather than locked inside custodial infrastructure or represented by synthetic derivatives.

2.3 Threat model assumptions

The security model assumes that:

- Neither the user nor GoMining is trusted individually with full spending authority; at least two signatures are required.
- GoMining is expected to operate honestly under normal conditions, but the system is designed for fault tolerance and user exit even in adversarial events.
- The user must protect their own key; compromise of the user key may enable unauthorized spending of pre-authorized limits.
- The recovery custody provider — selected per integration from a set of providers approved by GoMining (the wallet integrator may itself serve as the custody provider, subject to GoMining's

approval) — is expected to (a) honor the release delay policy, (b) remain operationally available, (c) protect the recovery signing capability from compromise, and (d) not collude with GoMining or with the user.

- The release delay is enforced by the custodian's policy and operational procedures, not by an on-chain timelock in the 2-of-3 script itself. The delay guarantee therefore depends on the custodian's honest execution of the policy.

Explicitly, the threat model considers:

- adversarial behavior by GoMining,
- forced closure or regulatory shutdown of GoMining,
- malicious user attempting to double-spend internally,
- leak or misuse of pre-authorized transactions,
- adversarial behavior by the recovery custody provider (premature release, unauthorized release, collusion with the user or with GoMining),
- unavailability or shutdown of the recovery custody provider, which would block the user from completing recovery,
- compromise of the recovery signing capability held by the custodian.

Two failure modes are outside the protocol's cryptographic enforcement scope and are mitigated only by operational and contractual controls on the custodian role:

- compromise of the user key combined with compromise of the recovery custodian, which yields a valid 2-of-3 witness (**User Key** + **Recovery Key**) without operator involvement;
- collusion between GoMining and the recovery custody provider, which yields a valid 2-of-3 witness (**GoMining Key** + **Recovery Key**) without user involvement.

Mitigations of custodian risk include selecting a reputable custody provider with appropriate operational controls (e.g., HSM-based key custody, segregation of duties, audit attestation) and contractual arrangements with legal recourse. These are mitigations, not cryptographic eliminations, of custodian risk.

When a wallet integrator also serves as the recovery custody provider, the user concentrates trust in a single party for both wallet key management and recovery custody. Compromise of that integrator yields a stronger attack surface than the reference configuration in which wallet, operator, and custodian are three independent parties. Integrators choosing this configuration accept additional trust concentration in exchange for operational simplicity.

The protocol does not require changes to Bitcoin consensus rules. User funds are held in native L1 UTXOs; the cumulative pre-authorization layer and the commercial merchant reconciliation layer are anchored to those L1 UTXOs.

3 Wallet Architecture

User wallets are the primary interface through which users hold BTC, authorize payments, and, if necessary, exit the ecosystem. Each wallet is represented on-chain as a 2-of-3 multisignature address, while the application layer manages key generation, storage, and recovery flows in a way that is usable for non-technical users. The architecture described in this section is the reference implementation provided by GoMining; the same flows are exposed via a wallet integration SDK so that third-party Bitcoin-capable wallets can implement instant-payment functionality natively under partnership with GoMining (Section 10).

3.1 Wallet creation

When a user creates a GoMining BTC wallet, three logical keys are established:

- **User Key** – generated and controlled by the user, stored in the user's wallet application.
- **GoMining Key** – generated and managed by GoMining in a hardened signing environment, used only to co-sign protocol-compliant transactions.
- **Recovery Key** – held offline by a trusted external custody provider, not used during normal operation. The recovery signing capability is released to the user under the conditions described in the Conditional Recovery mechanism (Section 7.2).

The corresponding 2-of-3 multisig script is constructed off-chain from the three public keys, and the address derived from this script becomes the user's wallet address. The address appears on-chain only once an output paying that script is included in a confirmed Bitcoin transaction. All user funds within the protocol are then held as UTXOs locked by such multisig scripts.

From the user's perspective, the wallet behaves like a standard non-custodial Bitcoin wallet: they see the total value of their multisig UTXOs, can receive BTC to their address, and can initiate payments. The difference lies in the fact that spending requires coordination between the User Key and the GoMining Key, with the Recovery Key reserved for the Conditional Recovery exit path (Section 7.2).

3.2 Transaction signing flow

In normal operation, a payment from a GoMining wallet is authorized in two steps:

1. **First signature: user.** The wallet application constructs the transaction. In the instant-payment flow, the user signs an updated cumulative pre-authorized PSBT reflecting the new total spend, using the User Key.
2. **Second signature: GoMining.** GoMining validates the request against internal rules: the sum of the user's spendable multisig UTXOs at the required confirmation depth (net of any amount currently pending settlement in an in-flight cumulative PSBT), risk controls, and the state of outstanding pre-authorizations. If the request is valid, GoMining adds its signature with the GoMining Key, producing a transaction that satisfies the 2-of-3 spending condition.

For ordinary on-chain withdrawals from the multisig address — spends not handled by the instant-payment subsystem, such as the user withdrawing BTC to an external Bitcoin address —

the user and GoMining jointly sign a standard Bitcoin transaction which is broadcast directly to the Bitcoin network.

For instant POS payments, the user signs an updated cumulative pre-authorized PSBT reflecting the new total spend. GoMining later co-signs this transaction as part of the batched settlement process (see Section 6). As an operational policy, GoMining's signer will only co-sign transactions that already carry a valid user signature; the 2-of-3 script itself does not enforce this ordering. Protocol-level protection against operator-alone spending derives from the Recovery Key being held by a separate custodian that does not co-sign with GoMining under any normal workflow. The sequence of user-signed PSBTs provides a cryptographic upper bound on the amount GoMining can settle for a given user.

3.3 Transaction construction details

This subsection specifies the concrete Bitcoin-level construction used by the protocol. Readers not interested in implementation details may skip ahead.

- **Script type.** Each wallet uses a 2-of-3 P2WSH (segwit v0, BIP 141) multisignature output, constructed from the sorted set of user, operator, and recovery public keys. The corresponding descriptor is `wsh(sortedmulti(2, pk_user, pk_gomining, pk_recovery))`. Addresses are bech32-encoded.
- **PSBT format.** All partially signed transactions use the BIP-174 PSBT v0 encoding, serialized as base64 for transport between the wallet and the operator.
- **Signature hash type.** User and operator signatures on settlement transactions use `SIGHASH_ALL`. Each settlement transaction is therefore fully committed at signing time: its inputs, outputs, and amounts cannot be modified by either party without invalidating the existing signature.
- **Replace-in-place pending PSBT.** At any moment a user has at most one active pending PSBT for settlement. When a new instant payment updates the cumulative amount, the operator constructs a new PSBT; the previous pending PSBT is superseded in the operator's internal records. Both PSBTs spend the same user multisig UTXO(s); they are therefore mutually conflicting Bitcoin transactions, and at most one of them can confirm on-chain.
- **Input selection and change.** Each settlement PSBT spends one or more of the user's multisig UTXOs, selected to cover the current cumulative amount and the network fee. Any residual value above the cumulative amount plus fee, and above the dust threshold, is returned as a change output to the same 2-of-3 multisig address; values below the dust threshold are absorbed into the network fee per Bitcoin standardness rules.
- **Confirmation depth for incoming UTXOs.** GoMining considers an incoming user multisig UTXO spendable, and may include it as input to a cumulative PSBT, only after it has reached at least 5 confirmations on the Bitcoin main chain.

3.4 User custody and limitations

The wallet architecture is designed to balance non-custodial control with the ability to provide instant operator-level payment acceptance to merchants, while on-chain settlement is subject to Bitcoin network conditions.

- **GoMining cannot spend user funds alone.** With only one of the three keys, GoMining cannot construct a valid transaction from the user's multisig address. Any spend requires either the User Key or the Recovery Key in addition to the GoMining Key.
- **The user cannot bypass protocol rules without recovery.** The user also controls only one live key. They cannot spend funds without either cooperating with GoMining or initiating the recovery process which comes with a delay and protocol-level consequences (removal from the instant-payment subsystem).
- **Recovery as an escape hatch.** If GoMining becomes unresponsive, refuses to co-sign, or is forced to shut down, the user can trigger the recovery flow and obtain access to the Recovery Key under the conditions described in Section 7.2 (operator-coordinated proof of settlement or fallback time-based delay). At that point, the user can spend their funds using the User Key plus Recovery Key without any GoMining participation.

This structure ensures that everyday payments can be fast and convenient while preserving a cryptographic exit option that returns spending capability to a user-controlled keypair (User Key + Recovery Key) in adversarial or failure scenarios.

4 Pre-Authorized Transactions

The protocol uses a cumulative pre-authorization model that allows instant Bitcoin payments without predefined spending limits. Instead of signing multiple denomination templates or a fixed maximum allowance, the user signs a growing cumulative spend value each time an instant payment is initiated.

This cumulative model gives GoMining a cryptographic upper bound on the amount it may settle on-chain for that user, while eliminating the need for predefined spending limits or pre-reserved funds. Residual UTXO value above the cumulative amount is returned to the user's multisig as a standard change output (see Section 3.3).

4.1 Definition

A **pre-authorized cumulative spend** is a partially signed Bitcoin transaction representing the total amount the user has authorized for the next on-chain settlement of their multisig UTXOs.

For every new instant payment of amount Δ:

$$S_{\text{new}} = S_{\text{prev}} + \Delta$$

the user's wallet signs a new pre-authorized transaction reflecting S_{new}. GoMining stores this latest signed value as the active pending settlement state, superseding earlier signed values in its operational records.

4.2 How it works

1. The user initiates a purchase of amount Δ.
2. The wallet sends a payment intent to GoMining indicating the merchant and the requested amount.

3. GoMining validates that the user's spendable multisig UTXOs cover the new cumulative amount $S_{\text{new}} = S_{\text{prev}} + \Delta$ at the required confirmation depth, computes S_{new}, and constructs a new unsigned PSBT reflecting this value.

4. The wallet receives the unsigned PSBT, verifies that the proposed S_{new} matches the wallet's own authoritative S_{prev} plus Δ, displays the payment details, and on user approval signs with the User Key.

5. The wallet returns the user-signed PSBT to GoMining; GoMining accepts and stores it as the new pending settlement state for the user. The operator's signature is added later, at batch settlement time (Section 6).

6. The instant payment is approved immediately.

GoMining retains the latest valid user-signed transaction for the cumulative amount that must eventually be settled. Because each new cumulative value strictly exceeds the previous one and each transition is cryptographically signed by the user, rollback and double-spend within the instant-payment subsystem are prevented: the user wallet refuses to sign a non-monotonic PSBT, and the operator cannot advance the cumulative value without a new user signature.

4.3 Security properties

- A malicious user cannot obtain an operator-accepted cumulative value that exceeds the sum of their confirmed multisig UTXOs, since the operator validates every proposed S_{new} against the user's confirmed UTXO set before co-signing.
- GoMining cannot spend more than the total cumulative amount the user has explicitly signed.
- No pre-reserved funds or fixed limits are needed.
- The user's on-chain settlement contribution is cryptographically bounded by the latest signed cumulative value. Any residual value of the selected UTXOs above this amount and the proportional network fee is returned to the same 2-of-3 multisig address as a standard change output, preserving user custody of the remainder.

This model is simpler, safer, and more operationally efficient than fixed-limit or multi-denomination pre-authorizations.

5 POS Payments and Instant Payment Flow

Instant payments are executed by updating the cumulative pre-authorization amount and validating that the user's confirmed multisig UTXOs are sufficient to cover the new total. No predefined limits or denomination templates are required.

5.1 Instant POS payment flow

The instant payment flow is designed so that GoMining never constructs a spend without explicit user approval. All cumulative spend updates are initiated by the user and finalized only after the user signs a PSBT that fully reflects the new state.

The flow is as follows:

1. The merchant's POS terminal generates a payment request (including amount, merchant identifier, and invoice or order metadata) and encodes it as a QR code or NFC payload.

2. The user scans the QR code with the wallet application and reviews the payment details (amount, merchant, context).

3. The wallet sends a payment intent to GoMining, indicating the user's wallet, the merchant, and the requested amount.

4. GoMining retrieves the user's current cumulative spend S_{prev} and the sum of the user's confirmed multisig UTXOs, computes the new cumulative value

$$S_{\text{new}} = S_{\text{prev}} + \Delta,$$

   and constructs a PSBT that reflects this updated total spend and the corresponding internal allocation to the merchant.

5. GoMining returns the unsigned PSBT to the user's wallet together with a human-readable summary of the payment.

6. The wallet displays the full payment information (including the new cumulative total and the merchant details) and asks the user to confirm. The user can either approve or reject the payment.

7. If the user approves, the wallet signs the PSBT with the User Key and sends the signed PSBT back to GoMining.

8. GoMining verifies the signature and consistency of the PSBT and, if valid, records the payment in the internal mempool and immediately returns an *Approved* response to the POS terminal. From the merchant's perspective, the payment is operator-accepted at the point of sale; on-chain confirmation occurs later, when the corresponding batch settlement transaction is included in a block.

At no point can GoMining increase the cumulative spend without a corresponding user signature. If GoMining attempted to construct a PSBT with a higher amount than expected, the wallet would display the inflated total and the user could simply refuse to sign. The signed PSBT is therefore the single source of truth for the user's authorized spending.

5.2 Internal mempool

The internal mempool is an operational structure used for tracking pending instant payments until the next settlement batch. It is not security-critical for user fund safety: cumulative authorizations are bounded cryptographically by user signatures, and the user wallet retains its own authoritative record of the latest signed cumulative value S_{prev}, rejecting any operator-proposed PSBT whose declared S_{prev} does not match. Internal-mempool consistency is, however, required for correct operator–merchant commercial accounting.

For each user, GoMining stores:

- the latest cumulative pre-authorized PSBT signed by the user,
- the list of pending merchant allocations derived from that cumulative value.

Each new instant payment advances the cumulative spend state:

$$S_{\text{prev}} \rightarrow S_{\text{new}}$$

The mempool maintains the mapping between the cumulative amount and the distribution of individual merchant payments. Since GoMining cannot modify the cumulative values without a valid user signature, the mempool does not require external auditability or cryptographic commitments for fund safety. Its role is operational: preparing data for the next settlement batch.

5.3 Change output handling

Because each settlement transaction spends the user's selected multisig UTXOs in full, any residual value above the latest signed cumulative amount and the proportional network fee is returned as a standard change output to the same 2-of-3 multisig address.

The cumulative pre-authorization model therefore:

- does not require an "umbrella" pre-authorization or a fixed spending limit,
- does not lock excess funds: any residual UTXO value is returned to the user's own multisig as a change output and remains immediately available,
- produces, per participating user per batch, one settlement contribution bounded by the cumulative signed value plus at most one change output back to the user's multisig.

5.4 Merchant experience

Merchants receive an instant *Approved* response from the GoMining operator at the point of sale once the user has signed the updated cumulative pre-authorization. This operator-issued approval is a payment authorization within the GoMining payment ecosystem and is distinct from Bitcoin network confirmation, which occurs later when the corresponding batch settlement transaction is included in a block.

Each per-user settlement transaction finalizes that user's cumulative authorized amount by paying it to an operator settlement address. These transactions are not constructed as direct per-merchant payouts; no on-chain output corresponds to an individual merchant payment.

Merchant payouts and reconciliation are handled separately by GoMining as part of its commercial settlement processes. This separation allows the protocol to maintain highly efficient Bitcoin settlement transactions while providing predictable and simplified payment acceptance for merchants.

As a result, merchants benefit from instant operator-level payment acceptance and simplified operational integration without needing to interact directly with Bitcoin transaction construction or batching mechanics.

Because merchant payouts are reconciled as commercial settlement from a GoMining-controlled output, merchants hold unsecured credit against GoMining between on-chain settlement and commercial payout. Merchant acceptance of instant payments therefore relies on GoMining's solvency and operational continuity in addition to the cryptographic correctness of the cumulative pre-authorization. The instant-UX property is preserved while user funds remain self-custodied at the 2-of-3 multisig up to the point of on-chain batch settlement.

6 Batching and On-Chain Settlement

Settlement is not a single on-chain transaction per batch. Instead, each participating user's cumulative authorization is settled through its own per-user Bitcoin transaction, and a group of such per-user transactions is aggregated into a single Child-Pays-For-Parent (CPFP) child transaction that pays the market-rate fee for the entire package. "Batching" in this protocol therefore refers to grouped propagation and CPFP-based fee aggregation, not to packing multiple user inputs into one transaction.

This design keeps each settlement transaction small, standard, and independently valid, while using a single child transaction to bound the per-batch fee cost and preserve miner economics.

6.1 Per-user settlement transaction

Each active user is associated with at most one pending settlement PSBT at a time, as described in Section 3.3. The transaction has:

- **inputs:** one or more of the user's multisig UTXOs, selected to cover the current cumulative authorized amount S_{new} and the network fee;
- **outputs:** a single output to the operator settlement address carrying S_{new}; plus, if the residual UTXO value exceeds the dust threshold, a change output back to the same 2-of-3 multisig address.

When a user makes multiple instant payments during a batching interval, the operator replaces the pending PSBT with a new one reflecting the updated cumulative amount (see Section 3.3). Only the final, latest user-signed PSBT for that pending interval is broadcast; earlier signed PSBTs are never transmitted, and because they spend the same user input they are mutually conflicting.

6.2 Operator internal accounting

Internally, the operator maintains a mapping between the cumulative amount authorized by each user and the list of merchant payment records that contributed to that amount. These records are used for commercial reconciliation with merchants and do not appear as outputs in the Bitcoin settlement transactions.

The operator's internal accounting layer is not a security-critical component for user funds: cumulative authorizations are bounded cryptographically by user signatures, and the user wallet verifies each proposed cumulative amount before signing. The accounting layer is an operational aggregation structure that determines when and how per-user settlement transactions are constructed and grouped.

6.3 Commercial reconciliation layer

Merchant payments recorded within the GoMining instant payment subsystem are tracked as part of GoMining's internal accounting and reconciliation layer. Each instant payment record contains the merchant recipient and the corresponding amount; together, these records represent commercial payment obligations between GoMining and participating merchants.

These merchant payment records are not mapped to individual outputs in the per-user Bitcoin settlement transactions. Instead, the on-chain flow transfers cumulative user-authorized amounts to an operator settlement address, from which merchant payouts are reconciled separately by GoMining.

This separation decouples the number of on-chain outputs from the number of merchants receiving payments: the number of outputs per settlement transaction depends only on the per-user structure (one operator output plus an optional change output). Merchant reconciliation occurs as commercial settlement according to agreements and operational policies defined by GoMining.

6.4 Batch aggregation via CPFP

When the operator decides to settle a group of per-user pending PSBTs, each user-signed PSBT is co-signed with the operator key and finalized into a standard Bitcoin transaction. Each such transaction carries a minimal base network fee.

The operator then constructs a single Child-Pays-For-Parent (CPFP) child transaction whose inputs are the operator settlement outputs of some or all of the newly finalized parent transactions, and whose output pays the aggregated value to the operator settlement address. The child carries a fee sized to bring the effective feerate of the parent-plus-child mempool package up to a market-competitive level.

The group of parent transactions plus the single child transaction forms a *settlement batch*: a connected mempool package whose inclusion economics are driven by the child's fee. All transactions in the package are standard, independently valid Bitcoin transactions.

Timing of settlement is policy-driven rather than on a fixed schedule: small per-user amounts may remain pending across several intervals, while larger amounts are included in the next batch. The protocol does not mandate a fixed batching interval; the operator determines when settlement is economically appropriate, subject to any commitments it makes to users and merchants.

6.5 Fee model and merchant economics

GoMining absorbs the Bitcoin network fees required to confirm settlement transactions, including the fee paid by the CPFP child described above. Users therefore do not pay a direct on-chain fee when making instant payments; these costs are recovered through merchant service fees.

Each per-user settlement transaction carries only a minimal base fee. The market-rate fee for the whole settlement batch is paid by the single CPFP child transaction. The effective on-chain cost per instant payment therefore decreases as more user settlements are included under the same child: the child's fee is amortized across the entire package rather than charged per user transaction.

The number of outputs in each per-user settlement transaction depends only on the per-user structure (one operator output plus an optional change output), independent of how many merchant payments that user made during the interval. The system can therefore support a large number of merchant payments without increasing the on-chain output count.

The CPFP mechanism is always applied, independently of which pool finds the block. When a block produced by the GoMining pool includes settlement transactions, the combined package fee accrues to the block finder (including third-party miners hashing on the pool), preserving the

per-block payout such miners would expect from the public mempool; when a block is produced by another pool, the same mechanism makes the package competitive for inclusion.

Parent and child transactions are held in an operator-side staging queue and submitted, via the Stratum V2 Job Declaration sub-protocol, to GoMining-operated mining infrastructure. At the time of writing, GoMining produces on average approximately two blocks per day, in which these transactions are included directly; for blocks produced by other mining pools, inclusion proceeds through standard public-mempool propagation. The operator may also broadcast to the public Bitcoin network at its discretion during infrastructure or network-capacity constraints.

Merchants pay a service fee to GoMining according to commercial agreements. This fee reflects payment processing, risk management, and merchant reconciliation services provided by GoMining rather than the direct construction of per-merchant outputs in settlement transactions.

6.6 Scalability considerations

Per-user settlement transactions are compact: their size depends on the number of user multisig UTXOs consumed and on whether a change output is produced, and is unaffected by the number of merchant payments the user made during the interval.

The main on-chain scalability driver is therefore the number of distinct users settled in a given batch, not the number of merchant payments. Standard Bitcoin mempool policy bounds the size of the connected parent-plus-child package (for example, the default limits on ancestor and descendant count for unconfirmed mempool chains); when a batch would exceed these bounds, the operator splits it into multiple independent packages, each with its own CPFP child.

The operator determines when to settle each user's pending PSBT based on cumulative amounts, elapsed time, and confirmation timing objectives, rather than on a fixed interval. This policy allows small accumulated amounts to remain pending across multiple cycles while larger amounts are settled in the next available batch.

In practice, settlement scalability is determined by the available Bitcoin block space, Bitcoin mempool policy limits on package size, and the operational thresholds used by GoMining to decide batch composition.

7 Conditional Recovery and Exit

The GoMining BTC Instant Payment Protocol provides users with a unilateral exit mechanism called *Conditional Recovery*, by which they can regain control of remaining multisig funds without GoMining's cooperation. The cumulative pre-authorization model integrates naturally with this mechanism: even if GoMining becomes unable or unwilling to co-sign transactions, the user can retrieve remaining BTC on the Bitcoin main chain through the recovery path, subject to the assumptions on the recovery custody provider stated in Section 2.3.

7.1 Recovery key design

The third key in the 2-of-3 multisig structure (**Recovery Key**) is not held online and is not used during normal operation. Instead, the recovery signing capability is securely held by a trusted external custody provider and released to the user under the conditions described in Section 7.2 — either an operator-coordinated proof of settlement or a fallback time-based safety delay, whichever occurs first.

This release process is designed to prevent malicious or premature exit attempts and to allow pending settlement operations to be completed before recovery access is granted.

The release delay is enforced by the custodian's policy and operational procedures; the 2-of-3 script itself does not contain an on-chain timelock. The strength of the delay guarantee therefore depends on the custodian's honest execution of the policy. Any legal or contractual arrangement with the custodian is a mitigation of custodian risk, not a cryptographic elimination of it. See Section 2.3 for the full custodian threat model and Section 10 for an architectural direction that would shift this delay from a policy-enforced to an on-chain-enforced property.

The provider does not participate in everyday transaction signing and cannot spend user funds independently as long as the custodian honors the release policy and is not compromised or in collusion. Its role is limited strictly to facilitating the user's unilateral exit from the system if GoMining becomes unavailable.

The goal is to allow a user, once the recovery release condition is met (Section 7.2), to satisfy the 2-of-3 spending condition with

$$\text{User Key} + \text{Recovery Key}$$

without requiring GoMining's cooperation.

7.2 Recovery request flow

A recovery request represents the user's intention to leave the ecosystem and regain direct unilateral control of funds. When the user initiates recovery, the request becomes visible both to GoMining and to the recovery custody provider.

Operator-side actions.

1. Instant payments are immediately disabled for the user; no new cumulative spend updates are accepted, and the internal-mempool state for the user is frozen.
2. The latest user-signed cumulative PSBT (if any) is co-signed by GoMining, finalized into a standard Bitcoin transaction, and broadcast as part of the next available settlement batch.
3. Once the settlement transaction has confirmed on-chain at the operator's required confirmation depth, GoMining transmits to the custody provider proof of settlement, consisting of the on-chain transaction identifier and the confirmation depth reached.

Custodian-side release flow. The custody provider releases the recovery signing capability to the user under one of two conditions, whichever occurs first:

- **Primary path (operator-coordinated).** On receipt of the operator's proof of settlement, the custody provider verifies on-chain that the referenced settlement transaction is included in a block at the required depth, and then releases the recovery signing capability. Under this path, the settlement transaction is already final on-chain when the recovery key is released; a subsequent recovery spend cannot conflict with outstanding settlement.
- **Fallback path (time-based safety).** If the operator's proof of settlement has not been received within a configured safety delay T since the recovery request, the custody provider releases the recovery signing capability unconditionally. This fallback ensures that operator unavailability, regulatory shutdown, or refusal to cooperate cannot indefinitely block user recovery.

At this point the user can combine their User Key with the Recovery Key to spend the remaining multisig funds independently on the Bitcoin network.

7.3 Exit and settlement finalization

Once a recovery request is active:

- the user cannot perform further instant payments,
- GoMining must complete, reconcile, or cancel any pending commercial payment obligations associated with the user's cumulative spend,
- the last user-signed cumulative spend value becomes the final user-authorized settlement amount to be settled on-chain before exit.

After this process completes, the user can independently spend the remaining multisig funds using the User Key and the Recovery Key, without GoMining's cooperation.

7.4 Race and revocation considerations

A user-signed cumulative PSBT cannot be unilaterally revoked once delivered to the operator. The only way an outstanding signed PSBT can be rendered moot is for the underlying user multisig UTXO to be spent through a different transaction — in practice, through the recovery path — before the operator broadcasts the corresponding settlement transaction.

In the primary recovery path, the settlement transaction is broadcast and confirmed before the recovery key is released; a race between settlement and a recovery spend is therefore impossible.

In the fallback time-based path, a race is theoretically possible: if a settlement transaction remains in the mempool at the moment the custodian releases the recovery key, the user's subsequent recovery spend and the outstanding settlement transaction both become valid and conflicting Bitcoin transactions. Bitcoin's standard double-spend resolution then applies: whichever transaction is mined first prevails; the other becomes invalid.

The fallback path is intended for operator-unavailability scenarios rather than routine recovery requests; the safety delay T is calibrated to exceed typical settlement confirmation times, so the race window is narrow in practice but not eliminated. Section 10 outlines an architectural direction that further reduces this window by shifting settlement verification to the custodian.

7.5 Security considerations

The cumulative pre-authorization model simplifies and strengthens the recovery process:

- GoMining cannot freeze funds beyond the recovery release window; once recovery completes (via the operator-coordinated path or the time-based fallback, see Section 7.2), recovery provides unilateral user exit.
- A user cannot overspend before recovery: each cumulative update is validated against the user's confirmed multisig UTXOs before the operator co-signs.
- The recovery release process — combining operator-side settlement of any pending instant-payment state and a time-based fallback safety delay — protects the system from combined internal double-spend + exit attacks.

- After exit, the user can spend the remaining multisig funds via User Key + Recovery Key on L1 without reliance on GoMining infrastructure.

The recovery mechanism constrains operator trust through the cryptographic 2-of-3 structure and keeps the protocol resilient to operator failure, while protecting the ecosystem from malicious user behavior.

8 Security Analysis

The cumulative pre-authorization model strengthens the security foundations of the GoMining BTC Instant Payment Protocol. The system is designed so that neither the user nor GoMining has unilateral control over funds, and each party is protected from adversarial behavior of the other.

This section focuses on the two primary threat scenarios relevant to the protocol: operator-side freeze attacks and malicious user attempts to overspend or manipulate instant payments.

8.1 Threat model overview

The protocol evaluates the following two adversarial cases as highest priority:

1. **Operator freeze attack:** GoMining becomes unresponsive, refuses to co-sign transactions, or attempts to block user withdrawals.
2. **Malicious user behavior:** A user attempts to exploit instant payments by overspending, double-spending inside the internal mempool, or manipulating the cumulative spend updates.

The protocol provides cryptographic protections against both attack classes; the strength of these protections is subject to the assumptions on the recovery custody provider stated in Section 2.3 — specifically that no two of {User Key, GoMining Key, Recovery Key} are simultaneously compromised or colluding.

8.2 Protection against GoMining freeze attacks

User funds at the multisig address are held directly on the Bitcoin main chain inside a 2-of-3 multisignature output. GoMining cannot move or freeze these funds permanently because it controls only one of the three keys.

If GoMining refuses to co-sign valid transactions, becomes inaccessible, or is forced to shut down, the user triggers the Conditional Recovery flow (Section 7.2). Through the dual release path described there — operator-coordinated proof of settlement or fallback time-based delay — the user obtains the Recovery Key or equivalent signing capability, enabling:

$$\text{User Key} + \text{Recovery Key} \;\Rightarrow\; \text{valid 2-of-3 witness}$$

This allows the user to unilaterally spend the remaining multisig funds without requiring GoMining.

Thus, even in a total operator failure scenario, the user can unilaterally recover control of the remaining multisig funds via the Conditional Recovery path, and the system gracefully degrades into standard on-chain Bitcoin spending.

8.3 Protection against malicious user behavior

The cumulative pre-authorization model is designed to prevent the principal classes of user-side fraud within the instant-payment subsystem, summarized below.

Each instant payment requires the wallet to verify a new cumulative value

$$S_{\text{new}} = S_{\text{prev}} + \Delta$$

proposed by the operator, and to sign a PSBT reflecting this verified value.

Security properties:

- **Overspending is prevented.** Even if the operator proposes a PSBT whose cumulative value exceeds the user's spendable multisig UTXOs, the wallet rejects such a PSBT before signing. Under honest operator behavior the operator additionally avoids constructing such PSBTs, returning an error to the user before PSBT construction when insufficient funds exist.
- **Rollback to an earlier cumulative state is prevented.** Since $S_{\text{new}} > S_{\text{prev}}$ and each update is cryptographically signed by the user, the wallet refuses to sign a non-monotonic PSBT and the operator cannot unilaterally revert to a prior state.
- **Double-spending within the instant-payment subsystem is prevented.** Each new payment advances the cumulative value; all cumulative PSBTs for a given input conflict at the input level, so at most one can be confirmed on-chain.
- **Batch settlement is bounded by the latest signed value.** The authoritative settlement value is the latest signed cumulative PSBT. GoMining is operationally committed to broadcasting this latest value; this commitment is enforced through operator policy and merchant agreements rather than by the Bitcoin protocol, since any earlier signed S_i remains an independently valid Bitcoin transaction. Broadcasting an earlier value does not endanger user funds but reduces the amount GoMining settles to itself, creating an accounting deficit against its own merchant payouts.

The cumulative sequence forms a monotonic, cryptographically signed record of user-authorized spending whose upper bound the operator cannot exceed and which the user cannot falsify.

8.4 Summary

The security guarantees of the protocol can be summarized as follows:

- GoMining cannot freeze funds beyond the recovery release window or confiscate BTC outright. While a recovery is in progress, funds at the user's multisig are temporarily unspendable through the normal co-signing path; once the recovery release condition is met (operator-coordinated proof of settlement or fallback time-based delay, see Section 7.2), Conditional Recovery restores unilateral user control.

- Overspending and double-spending within the instant-payment subsystem are prevented: each cumulative update is signed by the user and validated by the operator against the user's confirmed multisig UTXOs before co-signing.
- The protocol maintains safety even during network congestion, partial system outages, or operator malfunction.
- The cumulative pre-authorization model provides stronger and simpler guarantees than fixed-limit or multi-template schemes.

Together, these properties ensure a robust payment infrastructure with cryptographically constrained operator trust, preserving the non-custodial wallet model with a unilateral recovery path while enabling instant merchant acceptance.

Operational settlement infrastructure.

In addition to the cryptographic properties described above, GoMining operates supporting infrastructure to improve the reliability and predictability of settlement confirmation.

Each batch consists of per-user settlement transactions plus a single Child-Pays-For-Parent (CPFP) child transaction that spends the operator-settlement outputs of those parents and carries a fee sized to raise the effective package feerate to a market-competitive level. The CPFP child is always attached, including for blocks produced by the GoMining pool — its role is to preserve normal mining payouts for third-party miners hashing on the pool and to make the package attractive for inclusion in blocks produced by other pools.

The parent-plus-child package is held in an operator-side staging queue and submitted, via the Stratum V2 Job Declaration sub-protocol, to GoMining-operated mining infrastructure. At the time of writing, GoMining produces on average approximately two blocks per day, in which these transactions are included directly; for blocks produced by other mining pools, transactions reach miners through standard public-mempool propagation. The operator may also broadcast the package to the public Bitcoin network at its discretion during infrastructure or network-capacity constraints.

These mechanisms are operational optimizations rather than protocol-level trust assumptions: all settlement transactions remain valid, standard Bitcoin transactions that any Bitcoin node accepts and that can be broadcast publicly if required.

9 Comparison to Alternative Approaches

Bitcoin's global payment landscape includes multiple approaches to fast or low-friction transfers, each with different trade-offs in trust, complexity, custody, and L1 anchoring. The GoMining BTC Instant Payment Protocol represents a distinct and complementary model for enabling instant merchant payments using real L1 Bitcoin with non-custodial control and minimal operational overhead.

9.1 Comparison overview

Solution	Non-custodial	Instant	Real BTC L1	UX simplicity	End-user fees
Lightning (self-hosted node)	Yes	Yes	Yes (channel UTXO)	Complex	Low
Lightning (custodial wallet)	No	Yes	No (IOU)	Simple	Low
Custodial processors	No	Yes	No (IOU)	Simple	High
Liquid / sidechains	Federated	Yes	Synthetic BTC	Medium	Medium
Ark / experimental	Yes	Yes	No direct L1	Medium	Very low
GoMining Protocol	**Yes**	**Yes**	**Yes**	**Very simple**	**None direct**

9.2 Narrative comparison

Compared to custodial payment processors. Custodial crypto payment systems settle balances internally and hold full control over user funds. They provide instant operator-level acceptance but introduce full counterparty risk for the entire user balance, regulatory exposure, and lack of transparency. The GoMining Protocol achieves instant operator-level acceptance while keeping uncommitted user balances self-custodied at a 2-of-3 multisig with a Conditional Recovery exit path. Counterparty exposure to the operator is limited to amounts already committed through cumulative pre-authorization and pending commercial merchant payout, rather than the user's full balance. Users do not deposit funds into an operator balance sheet during normal operation; GoMining does not control user multisig funds unilaterally. After on-chain batch settlement, the settled amount is held under operator custody pending commercial merchant payout (see Section 6).

Compared to the Lightning Network. Lightning enables fast peer-to-peer Bitcoin transfers over a network of payment channels. A self-hosted Lightning node is fully non-custodial: channel funds live in 2-of-2 funding UTXOs on L1, and the user can always unilaterally close a channel back to L1. However, operating a self-hosted node requires liquidity management, inbound and outbound channel capacity, routing availability, and ongoing operational effort. Custodial Lightning wallets remove that operational burden but place user funds under the wallet provider's custody and expose users to counterparty risk for their full balance.

The GoMining Protocol occupies a different point in the design space. It eliminates routing complexity entirely — any user can pay any merchant without channel topology, inbound liquidity, or routing availability — while keeping uncommitted user funds self-custodied at a 2-of-3 multisig on L1, rather than placing the full user balance under a custodial wallet provider. Instant payment acceptance is validated by cumulative pre-authorization rather than by successful channel routing.

Compared to Liquid and other sidechains. Sidechains rely on federated or permissioned networks and wrap BTC into synthetic representations. Settlement occurs off-chain and depends on the honesty of federation members. GoMining uses no synthetic BTC: funds remain in L1 UTXOs under user control. Settlement batches are regular Bitcoin transactions without any bridging or federation trust.

Compared to emerging off-chain protocols (e.g., Ark). Ark designs enable non-custodial off-chain payments but require special coordinators, periodic rounds, and specialized wallet

behavior. The GoMining Protocol provides a simpler cumulative model: every instant payment updates a monotonic user-signed cumulative spend, eliminating the need for rounds, special scripts, or complex output structures.

9.3 Positioning

The GoMining Protocol is best viewed as an **instant Bitcoin settlement infrastructure** tailored for merchant payments, combining:

- real L1 Bitcoin custody,
- instant UX for users and merchants,
- low effective on-chain cost per payment via CPFP-amortized batching,
- simple merchant integration via SDK,
- trust-reduced architecture with a cryptographic recovery exit path.

Rather than competing with Lightning or sidechains, the protocol complements the broader Bitcoin ecosystem by enabling high-volume retail transactions without sacrificing user self-custody of uncommitted balances, the cryptographic exit path, or user experience.

10 Conclusion

The GoMining BTC Instant Payment Protocol introduces an alternative and highly practical approach to instant Bitcoin payments. By combining native L1 Bitcoin custody with a cumulative pre-authorization model, the protocol enables instant operator-level payment acceptance at the point of sale, while retaining a non-custodial wallet model with unilateral user exit and lowering the effective on-chain cost per instant payment via CPFP-amortized batching.

The architecture is built around three core principles:

- **Real Bitcoin on L1.** Uncommitted user funds remain in standard Bitcoin P2WSH multisig outputs, without synthetic BTC representations, sidechains, or wrapped tokens.
- **Instant, predictable UX.** Merchants receive an operator-level *Approved* response at the point of sale, with no need for routing, channel capacity, liquidity management, or specialized hardware.
- **Non-custodial wallet with unilateral exit.** No single party can spend funds held at the 2-of-3 multisig unilaterally. The Conditional Recovery mechanism provides the user with a unilateral exit (see Section 7.2 for the dual release path), with a time-based fallback that ensures recovery cannot be indefinitely blocked even in operator-failure scenarios.

The cumulative pre-authorization design provides cryptographic bounds on settlement correctness while eliminating the need for predefined spending limits or denomination templates. Each active user's cumulative authorization is settled through its own compact per-user Bitcoin transaction to an operator settlement address, and a group of such transactions is aggregated into a single Child-Pays-For-Parent child transaction that carries the market-rate fee for the whole package.

Merchant reconciliation is handled separately as part of GoMining's commercial settlement processes. The number of on-chain outputs per user is independent of the number of merchant payments that user made during the interval.

At the same time, the protocol remains compatible with existing POS infrastructure through a lightweight SDK and can be integrated without changes to merchant hardware or user wallet fundamentals.

The result is a trust-reduced, scalable, and operationally simple infrastructure layer that brings real Bitcoin to everyday commerce, bridging the gap between decentralized money and instant retail payments.

Future work

The foundation presented in this paper is sufficient to deliver a fully functional instant-payment system backed by L1 Bitcoin security. A number of architectural directions are envisioned for future versions of the protocol:

- **On-chain-enforced recovery delay via Taproot.** Migrating the multisig construction from P2WSH `2-of-3 sortedmulti` to Taproot (P2TR, BIP-341) opens the possibility of expressing the recovery path through a script-path leaf with a relative timelock (`OP_CSV`). Under such a construction, the recovery delay would be enforced by Bitcoin consensus rather than by the custodian's policy: even a compromised or malicious custodian could not produce a valid recovery spend before the timelock elapses. This would shift the delay guarantee from a trust assumption on the custodian to a consensus-level invariant. The normal-path spend (user + operator) could in addition use MuSig2 key-path signing for improved on-chain efficiency and privacy.
- **Threshold-distributed recovery custody.** Splitting the recovery signing capability across multiple independent custody providers under a threshold scheme (m-of-n) would reduce single-custodian compromise risk: compromise of fewer than m providers would not yield a valid recovery signature.
- **Wallet integration SDK and partner program.** A public SDK and partnership program enabling third-party Bitcoin-capable wallets — multi-chain consumer wallets (e.g., Trust Wallet) and Bitcoin-native wallets (e.g., Sparrow, Electrum) — to integrate the instant-payment flow without users switching to a separate GoMining wallet application. The wallet integrator provides the user's public key through the SDK; the 2-of-3 multisig is then constructed against GoMining's co-signing key and a recovery custody provider from GoMining's approved set (the integrator may also serve as the custody provider, subject to GoMining's approval and partnership terms). The wallet retains custody of the user's key throughout, signs cumulative PSBTs locally, and surfaces the recovery flow to the user.
- **Custodian-mediated settlement verification.** Shifting more of the recovery release verification to the custody provider — for example, by having the custodian directly verify the co-signed settlement PSBT or broadcast it itself — would further reduce reliance on operator coordination during recovery. Under such a design, recovery release could proceed even in partial-operator-failure scenarios while preserving the invariant that no recovery spend conflicts with an outstanding settlement (see Section 7.4).
- **Transparency and auditability.** Optional public commitments to cumulative authorization values (e.g., periodic Merkle root publication) would allow third parties to verify operator behavior without inspecting internal accounting state.

- **Interoperability.** Integration paths with other Bitcoin payment rails, including BOLT12 and emerging silent-payment standards, are open directions.

Acknowledgments

The author would like to thank *Daniil Dmitriev* for valuable feedback and for proposing the improvement to the transaction authorization model, which significantly contributed to the final cumulative pre-authorization design presented in this paper.

Appendix A Cumulative State Verification Model

In the cumulative pre-authorization architecture, the security of user funds does not depend on the integrity of the operator's internal mempool. The cryptographic bound on the amount the operator can settle for a given user is provided directly by the sequence of user-signed cumulative spend values; the wallet retains its own authoritative record of the latest signed value and rejects any operator-proposed PSBT whose declared S_{prev} does not match. Internal-mempool consistency is required for correct operator–merchant commercial accounting, not for user fund safety.

A.1 Cryptographic state

For each instant payment the operator proposes a new cumulative value

$$S_{\text{new}} = S_{\text{prev}} + \Delta$$

in an unsigned PSBT; the user's wallet verifies this value against its own authoritative S_{prev} and signs the PSBT. This produces a monotonically increasing sequence:

$$S_0 < S_1 < S_2 < \cdots < S_n$$

where each transition is authorized by a user signature. Monotonicity on the wallet side is enforced by the wallet refusing to sign any non-monotonic PSBT; monotonicity on the operator side is preserved by honest retention of the latest signed value. The operator cannot advance the sequence without a new user signature, and the user cannot revert to an earlier signed value once superseded.

A.2 Operator limitations

Regardless of operator state, GoMining cannot:

- invent additional instant payments (no valid user signature for a new S),
- inflate the cumulative value beyond what the user signed (signature mismatch on the constructed PSBT),
- reorder or remove payments in a way that exceeds the latest signed value (the cumulative bound is fully determined by the user-signed S_n),
- settle more than the highest user-approved value.

Broadcasting an earlier signed S_i instead of the latest S_n remains technically possible, as all signed cumulative PSBTs for the same input are individually valid Bitcoin transactions. GoMining is operationally committed to broadcasting the latest value; this is an operator policy rather than a Bitcoin-level enforcement and does not endanger user funds.

Thus, the internal mempool is an operational structure and not security-critical for user fund safety: cumulative amounts are bounded cryptographically by user signatures, and the wallet cross-checks each operator-proposed S_{prev} against its own authoritative record. Its consistency is required for correct operator–merchant commercial accounting but does not require external publication or on-chain cryptographic commitments.

A.3 Settlement correctness

Final on-chain settlement is upper-bounded by the most recent value signed by the user:

$$S_{\text{final}} \leq \max(S_i).$$

The operator cannot settle any amount above this bound. Under honest operator policy, the operator broadcasts exactly $S_{\text{final}} = \max(S_i)$; settlement at a lower S_i remains technically possible but is disincentivized because it reduces the operator's own settled inflow relative to its merchant payout commitments. The user can verify the settled value directly from the on-chain settlement transaction, without relying on the operator's internal logs.

A.4 Auditability

Because the cumulative state is fully determined by the user-signed sequence of PSBTs, and settlement is strictly bounded by this sequence, the system provides per-user cryptographic verifiability: each user can independently verify the cumulative amount settled for them on-chain against their own retained sequence of signed cumulative PSBTs. Third-party auditability across users requires additional optional commitment mechanisms; this is discussed as a future direction in Section 10.

Appendix B Protocol Flow Diagrams

This appendix presents sequence diagrams for the key flows of the GoMining BTC Instant Payment Protocol. The diagrams reflect the cumulative pre-authorization model in which GoMining constructs PSBTs, while the user authorizes each cumulative spend update by explicitly signing or rejecting it.

B.1 Instant POS Payment Flow

Participants:

- POS Terminal (merchant)
- User Wallet
- GoMining Operator

Flow description

1. The POS terminal sends an initialization request to GoMining containing the merchant identifier and the requested amount Δ.
2. GoMining creates an internal payment record, attaches metadata (e.g. payment identifier, expiry), and returns payment data to the POS.
3. The POS encodes this payment data as a QR code (or NFC payload) and presents it to the user.
4. The user scans the QR code in the wallet application and reviews the basic payment details.

5. The wallet sends a payment intent to GoMining indicating the user, the referenced payment, and the amount Δ.

6. GoMining retrieves the user's latest cumulative spend value S_{prev}, computes

$$S_{\text{new}} = S_{\text{prev}} + \Delta,$$

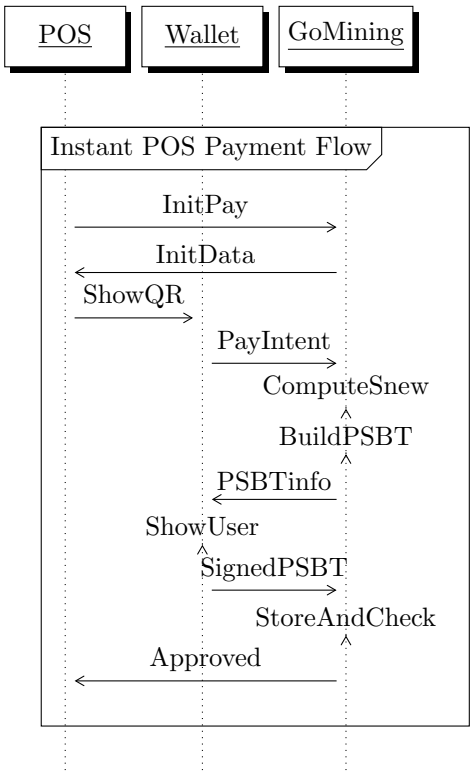
verifies that the user's confirmed multisig UTXOs cover S_{new}, and constructs an *unsigned* PSBT reflecting the new cumulative spend and the allocation to the merchant.

7. GoMining returns the unsigned PSBT together with a human-readable summary of the payment to the user's wallet.

8. The wallet displays the full payment information (merchant, amount, and updated cumulative total) and asks the user to confirm.

9. If the user approves, the wallet signs the PSBT with the User Key and sends the signed PSBT to GoMining.

10. GoMining verifies the signature and consistency of the PSBT, records the payment in the internal mempool, and immediately returns an *Approved* response to the POS terminal.



B.2 Cumulative State Update Flow

Concept

The cumulative state for a user is a strictly increasing sequence of total spend values $(S_0, S_1, \ldots, S_n)$, where each transition $S_{\text{prev}} \to S_{\text{new}}$ is explicitly authorized by a user signature on a PSBT constructed by GoMining.

Flow description

1. GoMining takes the previously signed cumulative value S_{prev} and a new requested amount Δ.
2. GoMining computes $S_{\text{new}} = S_{\text{prev}} + \Delta$ and constructs an unsigned PSBT for S_{new}.
3. The unsigned PSBT is sent to the user's wallet.
4. The user reviews the updated cumulative total and either signs or rejects the PSBT.
5. If signed, GoMining stores the signed PSBT as the new authoritative cumulative state for that user until the next payment.

Sequence diagram