pSymm

Private, Symmetrical Trading

Main Idea

Trade derivative assets privately, without the need to trust the counterparty.

- Deposit USDC
- Trade
- Withdraw

Hence the name - **p**rivate, **Symm**etrical trading.

Trading sessions follow a set of rules enforced by validators.

Roles

- **Trader:** The user, running in web UI environment.
- **Solver:** A backend app filling trader orders (e.g., by accessing a CEX).
- **Validator:** A SettleMaker voter (more on this later).

Components: Custody

A concept of a trading "room" between Trader and Solver.

- A Custody holds parties' collateral until the end of the trading session.
- In the end, the collateral is "rebalanced" based on parties' PnL.
- All trading messages are signed:
    - Each message contains a hash of all previous messages.
    - Includes an ECDSA signature of the current message.
- Trading session messages are not public, unless there is a dispute.

Components: SettleMaker

An onchain dispute settlement system.

- Validators pass KYC/AML, deposit collateral, and vote on disputed trading sessions.
- The vote is not opinion-based, but based on trading rules (e.g., a solver should fill with the same price as quoted).
- To dispute a trading session, one of the parties must reveal the whole trading session messages.
- Validators who don't vote with the majority lose collateral.
- Voting is batched in weekly resolutions.

Design Inspiration

- The idea to hide collateral is similar to Tornado Cash, except that the amount is arbitrary. (Similar to Toadnado project).
- **Core privacy feature:** No one can trace the nullifier hash (used on withdraw) to the commitment (deposit).
- The contract uses SNARKs to facilitate deposit and withdraw – to verify the knowledge of the secret for the commitment.

Circuits

We use several circuits to allow using commitments as collateral:

- **ATC (address to custody):** Deposit funds to pSymm - prove the commitment matches onchain funds.
- **CTC (custody to custody):** Split a commitment into two new ones.
    - Allows complete anonymization by breaking the link between original deposit/funds/address and new commitments.
    - Verifies the sum matches the original commitment.
    - Usable for privacy and PnL rebalance at the end of a trading session.
- **CTA (custody to address):** Withdraw funds using a nullifier.
    - Includes a timelock (few minutes) to prevent malicious collateral withdrawal if a trade goes unprofitable.

The Complete Flow (1/2)

1. Trader finds a Solver in the Party Registry onchain.
2. Trader connects to a Solver's websocket.
3. Trader asks for the instrument list.
4. **Logon:**
    - Trader makes a deposit (creates a commitment), sends Logon (with deposit proof) to solver.
    - Solver verifies deposit, makes a deposit, sends Logon.
5. Trader sends quote requests, orders; Solver responds.
6. Each message includes a hash of all previous messages and the sender's ECDSA signature.

The Complete Flow (2/2) - Withdraw

Two options:

- **a) Dispute on PnL:**
  - Use SettleMaker.
  - The wrong party gets slashed.
- **b) No Dispute:**
  - Losing party uses CTC to split its collateral into (PnL + rest of collateral).
  - Losing party sends private data of the PnL commitment to the counterparty.

Counterparty can withdraw its collateral and PnL to any address.

Idea: Add a circuit to merge commitments?

Thank you for listening

I'm happy to hear any thoughts, questions, or suggestions you might have