VIPER: An Intent-Based, Privacy-Preserving Cross-Chain Liquidity Network

"Privacy is Freedom."

Abstract

VIPER is a censorship-resistant DEX network that connects Zcash-grade privacy to any chain using **intent-based execution**, **shielded settlement**, and **cross-chain verification**. Users sign **private intents** describing desired outcomes (e.g., *swap USDC on Base → receive SOL on Solana*). Off-chain **solvers** compete to fill these intents at the best all-in price (swap + bridge), then settle through **RAILGUN** shielded pools and **Relay** light-client proofs—without public mempool leakage or linkable addresses. A native token **$VIP** funds solver liquidity and relayer gas via an on-chain treasury, bootstrapping execution quality while keeping the protocol credibly neutral (no VC/KOL allocations; LP seeded at launch; liquidity locked; contract renounced).

1 Motivation

Public mempools, doxxing gas top-ups, and deposit↔withdrawal linkability break financial privacy and enable MEV extraction. VIPER removes these failure modes by:

1. **Intents** instead of raw transactions,
2. **Private routing** via shielded pools + private RPC,
3. **Proof-based finality** across chains, not breadcrumbs,
4. **Market-based execution**—solvers compete, users get best price,
5. **Selective disclosure** via view keys when needed.

2 System Overview

2.1 Actors

- **User**: Signs an encrypted intent specifying outcome constraints.

- **Relayer**: Submits source-chain deposits via private RPC; sponsors gas with a paymaster.
- **Solver**: Off-chain agent that prices routes, quotes all-in fills, executes swaps/bridges, and provides proofs.
- **Verifier**: On destination chain, verifies zk light-client proofs from **Relay** to authorize shielded credits.
- **Treasury**: On-chain $VIP module allocating liquidity lines to approved solvers and funding paymaster budgets.

2.2 External Providers

- **NEAR Intents (or equivalent intent layer)**: encrypted intent messaging + matching surface.
- **RAILGUN**: shielded pools (Groth16-style zk circuits), view keys, privacy-preserving transfers.
- **Relay**: cross-chain verification using light-client proofs for source-side spend/commit inclusion.

3 Intent Lifecycle

3.1 Intent Format

```
I = Enc_pkM( {
  user_pub: P_u,
  src:  {chain: C_s, asset: A_s, amount: x},
  dst:  {chain: C_d, asset: A_d, min_amount: y_min},
  slippage_bps:   ,
  deadline: t_exp,
  privacy_flags: {shield_dst: bool, stealth_addr: bool},
  fee_prefs: {gas_sponsor: bool, max_fee: f_max},
  nonce: n
})
sig = Sign_skU(Hash(I_plaintext))
```

3.2 Quoting & Commitment

```
Q_i = { price_out: y_i, fee: f_i, route_digest: H(R_i), ttl: t_i } with
   sig_i
```

3.3 Source-Side Shielded Deposit

User or relayer deposits into RAILGUN pool, emits commitment, generates spend auth.

3.4 Cross-Chain Proof

Relay verifies zk spend proofs cross-chain.

3.5 Destination Settlement

Funds received shielded or stealth; gas sponsored.

4 On-Chain Architecture

4.1 Contracts (per chain)

- ShieldedPool (RAILGUN): commitments, nullifiers, zk verify.
- RelayVerifier: proof verification for cross-chain inclusion.
- VIP Treasury: manages solver liquidity lines, funds paymaster.
- Paymaster: gas sponsor with policy guardrails.
- IntentAnchor (optional): records hashes for arbitration.

4.2 Solver Vaults

Capital-efficient vaults managed by solvers with caps.

5 Economic Design

5.1 Quote Selection

$$\max_i \; y_i \quad \text{s.t. slippage} \le \sigma, \; t \le t_{exp}, \; \text{privacy flags satisfied} \tag{1}$$

5.2 Solver PnL

$$\Pi_i = (f_{route} + \text{rebates}) - (\text{inventory cost} + \text{gas}_{net}) \tag{2}$$

6 Token Utility

Token Utility $VIP

- **Gasless Transfers**: holders transact privately with zero gas cost, sponsored via relayer paymasters.
- **Fee Rebates**: execution fees are rebated in $VIP, rewarding active users.
- **Relayer LP Role**: by pledging 100,000 $VIP, holders become relayer LPs, supplying liquidity and earning rewards.

7 Privacy & Security

Protections: no mempool, zk commitments, proof-based bridging, stealth addressing, gas privacy.

8 Formal Flow (Pseudocode)

```
User:
  I_plain = {...}
  I = Enc_pkM(I_plain)
  sig_u = Sign_skU(Hash(I_plain))
  send_to_intent_layer(I, sig_u)
...
```

9 Interfaces

9.1 Intent API

```
type Intent = {...}
```

9.2 Solver Quote

```
type Quote = {...}
```

10 Governance & Upgradability

Genesis posture: no admin keys, LP locked. Governance: solver allowlist, paymaster budgets, supported chains/assets, disclosure policies.

11 Compliance Posture

Default: private by design; view keys for selective disclosure.

12 Security & Audit Plan

Audits: RAILGUN integration, Relay assumptions, Paymaster, solver vaults. Bug bounty program.

13 Performance & Metrics

Metrics: anonymity set size, slippage vs. public routers, time-to-settle,

14 Roadmap

- Phase 0: Testnet mock Relay + RAILGUN fork, sandbox, 23 solvers.
- Phase 1: Mainnet Beta on 2 L2s + Solana; SLR lines to capped solver set.
- Phase 2: Additional chains, generalized intents; stealth UX; mobile signer.
- Phase 3: Solver marketplace with bonding, scorecards; decentralized matching relays.

15 Conclusion

VIPER delivers best-execution liquidity with CEX-grade privacy through intents, zk-shielded settlement, and proof-driven bridging. Privacy is the foundation of unstoppable money.

A Glossary

Intent, Solver, Shielded Pool, Relay, Paymaster, View Key.

B Disclaimers

Components modular; provider integrations follow upstream security models.