Zecpass

The pass from Zcash to Robinhood Chain

Version 1.0, September 2026

Abstract

Zcash is the only widely traded asset whose transaction history can be genuinely private. Robinhood Chain is where tokenised equities and the Pons token launcher live, and it is as transparent as every Ethereum rollup. No road existed between the two that kept the Zcash side private and the Robinhood side free of custody. Zecpass is that road. It composes two settlement networks that already run, NEAR Intents and Relay, with a small hub contract on Robinhood Chain that turns incoming ETH into $ZECPASS for a beneficiary chosen in advance. The design goal is a single sentence that can be defended: nothing on Robinhood Chain links the address that receives $ZECPASS to the ZEC that paid for it. This paper states what that sentence rests on, the exact trusted surface, the token whose fees fund the infrastructure, and what the design leaves open.

1. The problem

Every EVM chain publishes who paid whom. That is the property that makes on-chain markets auditable, and it is also the property that makes them unusable for anyone whose entry should not be a public event: a

trader whose treasury is watched, a builder whose wallet is known, a person who simply does not want a permanent record attached to a purchase.

The privacy tools that exist inside EVM chains, mixers and shielded pools written in Solidity, have two weaknesses. They are small, so the anonymity set is small and the exit is often the most traceable transaction on the chain. And they carry a regulatory weight that follows any protocol that touches them.

Zcash has neither weakness. Its shielded pool is large, has ten years of history, and its privacy is a property of the base layer rather than a contract. In 2026 shielded usage passed half of all transactions and the asset was re-rated by the market accordingly. What Zcash does not have is a way to reach the places where things happen on EVM chains.

The obvious route, sell ZEC on a centralised exchange and withdraw ETH, hands the exchange a complete record. The next obvious route, a cross-chain swap into an EVM address, leaves the swap's destination address visibly funded by a swap that visibly came from Zcash, which is fine, and then the user buys the asset they wanted from that same address, which is also fine. The gap is small: what is missing is a product that makes this two-step road one step, keeps it non-custodial, and is honest about which step is which.

2. The road

A pass has five stations. Only the third and fourth were written by Zecpass.

Station one: the shielded wallet. The user holds ZEC in the shielded pool, in Zodl or an equivalent wallet. They ask Zecpass for a pass by giving three values: an amount, a Robinhood Chain address that should receive $ZECPASS (the beneficiary), and a transparent Zcash address for refunds.

Station two: NEAR Intents. Zecpass asks the 1Click API for a quote: ZEC in, ETH on Arbitrum One out, recipient the Zecpass relayer. The answer contains a fresh transparent Zcash deposit address and a

deadline. The user sends the ZEC there from their shielded balance. On the Zcash chain this is a deshielding transaction: it shows an amount arriving at a transparent address and nothing about the sender. NEAR Intents' solvers fill the intent and pay ETH on Arbitrum One. Measured settlement time is about seven and a half minutes.

Station three: the relayer. A single process that watches NEAR Intents for the settlement. When the ETH lands, it does one of two things, described in section 4: it delivers immediately from a vault on Robinhood Chain, or it sends the ETH onward through Relay.

Station four: Relay and the pass. Relay moves ETH from Arbitrum One to Robinhood Chain in about one block. Its recipient is not the relayer and not the beneficiary but the beneficiary's *pass*: a contract address on Robinhood Chain derived from the beneficiary address alone, which exists as a destination before any code is deployed at it.

Station five: the hub. Anyone, in practice the relayer, calls `sweep(beneficiary)` on the Zecpass hub. The hub deploys the pass if it has no code, pulls its ETH, keeps a fee of half a percent for the treasury, buys $ZECPASS on the Pons curve, and transfers the tokens to the beneficiary. The pass is empty again and can be used again.

The whole road, on a good day, is eight minutes. The user has done one thing: sent ZEC from a shielded balance to an address.

3. The privacy claim, precisely

Walk the road backward from the beneficiary and record what each observer can establish.

An observer of Robinhood Chain sees the beneficiary receive tokens in a transaction that pulled ETH from a pass, and sees that ETH arrive at the pass from Relay. They can look at Arbitrum One and see the Zecpass relayer pay Relay that amount, and see the relayer receive it from a NEAR Intents solver seconds earlier. They can query NEAR Intents' public explorer and find the swap: so much ZEC, deposited at such a transparent address at such a time, paid out as so much ETH. They can

look at Zcash and see the deshielding transaction that funded that deposit address.

And there the walk stops. The deshielding transaction has no visible sender. The shielded pool's proofs guarantee that the ZEC that left it cannot be matched to the ZEC that entered it, to any address, or to any previous transaction. The observer has reconstructed the entire road and learned that *some shielded balance* paid for it.

So the claim is exactly this: the road is fully traceable from the beneficiary back to a deshielding transaction, and not traceable from that transaction to a person. The Robinhood Chain address has no owner that any chain can name.

Three things the claim does not say.

It does not say the beneficiary's later behaviour is private. Everything the beneficiary does after receiving tokens is public in the ordinary way.

It does not say amounts and timing are hidden. A user who deshields exactly one ZEC every Monday morning and whose pass receives the matching ETH every Monday morning can be correlated by anyone, without any Zecpass involvement.

It does not say the relayer is blind. The relayer necessarily learns, at settlement, which NEAR Intents deposit address paid which pass. This is the same link an observer of Arbitrum One already has, so the relayer's knowledge adds nothing to an adversary's, except the refund address, which is why the refund address should be fresh.

4. Two delivery modes and the trusted surface

Zecpass has one trusted party, the relayer, and its trust is bounded differently in two modes.

Bridge mode is the default. After NEAR Intents settles on Arbitrum One, the relayer signs one transaction: a Relay deposit whose recipient is the user's pass. Between the settlement and that transaction, the relayer's key holds the user's ETH. The window is seconds, the amount is one order, and every transaction involved is public. A relayer that stole would

be caught by the first person to read the stats page. The relayer then calls `sweep`, which anyone could call instead.

Instant mode exists to remove the Relay hop from the user's wait. If the hub's vault holds enough ETH and the order is within the caps, the relayer calls `deliver(orderId, beneficiary, amount)` the moment NEAR Intents settles; the hub buys from its own vault and the user receives tokens one block later. The Arbitrum ETH is then bridged to the hub to refill the vault. Here the relayer is trusted not to call `deliver` for settlements that did not happen. The contract bounds that trust: each order id can be delivered once, no delivery exceeds `maxPerOrder`, no UTC day exceeds `dailyCap`, and the vault is the treasury's capital, never a user's.

Two facts hold in both modes. First, the hub never holds user tokens across transactions. Second, the beneficiary can always call `withdraw` and receive the raw ETH in their pass, regardless of the owner, the relayer, a pause, or whether the token exists. There is no function that stops it. A pass is never a trap.

5. The hub and the passes

The hub is one contract, about three hundred lines, without inherited frameworks, deployed once on Robinhood Chain. It holds the vault, the fee, the caps, the token address, and a reference to a market adapter that knows how to buy on Pons: on the bonding curve first, and on the Uniswap V4 pool after graduation, including the moment in between when the curve is full and the pool does not yet exist.

A pass is a twenty-line contract deployed by the hub with CREATE2, where the salt is the beneficiary's address. Its address is therefore a pure function of the hub and the beneficiary, computable by anyone, valid as a destination before deployment. It has a `receive` and two functions that only the hub can call, both of which move its entire balance: one into a sweep, one to the beneficiary. It has no owner and no upgrade.

The consequence worth stating is that Zecpass does not need to be the only thing that pays a pass. Relay pays it today. A NEAR Intents solver can pay it directly once solvers quote Robinhood Chain, and the relayer's

probe for that route runs every ten minutes; the day it returns a quote, new orders use the pass as the NEAR Intents recipient, the Arbitrum leg disappears, and the relayer holds nothing at all. A friend can pay it. The hub does not ask where the ETH came from.

6. The token

$ZECPASS is launched on Pons V2, the token launcher of Robinhood Chain, as a plain bonding-curve token: a billion units, all on the curve, a virtual reserve of 1.68 ETH, graduation to a Uniswap V4 pool at 4.2 ETH paid in, a two percent creator tax to the treasury. There is no team allocation and no schedule. The launch buy, whatever its size, is public in the launch transaction.

The token's purpose is to be the thing a pass buys, and to fund the road. Two streams reach the treasury: the creator tax on every trade, and the half-percent fee the hub keeps on every pass. The treasury spends them on the relayer's gas on two chains, the machine that runs it, and the vault that makes instant mode possible. The vault is the only capital at risk in the system, and its size is a decision about how much instant capacity to offer, taken by hand.

This is a deliberately small economy. It pays for itself at a modest volume and has no mechanism that needs price to go up. Whether the treasury should buy $ZECPASS with what is left over is a question for the community that forms around it, not a promise in this paper.

7. What is left open

Stocks. Robinhood Chain's reason to exist is tokenised equities, and the hub's market adapter could as easily buy NVDA as $ZECPASS. Every rail in this paper would carry it. Whether a product whose input is shielded ZEC and whose output is a tokenised share of a listed company is a product anyone should ship is a regulatory question, and Zecpass version one answers it by not shipping it.

The direct rail. NEAR Intents lists ETH on Robinhood Chain as an asset today without a solver behind it. When one appears, the relayer stops

holding anything and the trusted surface in bridge mode becomes empty. The code path exists and is probed; it has never been exercised.

Aggregation. Passes are per beneficiary, and each sweep is its own trade on the curve. Batching sweeps would reduce curve impact and slightly blur timing correlation. It would also introduce a moment when the hub holds several users' ETH, which the current design avoids on purpose.

Amount privacy. Nothing here hides amounts. A Zcash-native approach, in which the user deshields a round amount to their own transparent address first and then splits, is available to any user and is outside Zecpass.

8. Summary of guarantees

- The Zcash side is as private as the shielded pool: the deshielding transaction reveals amount, destination and time, and nothing else.
- Zecpass holds no user value on Robinhood Chain outside a single transaction; ETH in a pass is recoverable by the beneficiary alone, always.
- The relayer's trust is one order in bridge mode, and the vault's caps in instant mode.
- Fees are bounded on chain: at most two percent to the hub, set to half a percent.
- The contract is verified, the tests run against the live launcher on a fork, and every number on the site is read from the chain or the rails, never estimated.

Zecpass is a short road. Its value is that it is short, that each station is someone else's well-tested product or a contract you can read in ten minutes, and that it says out loud where the trust is.

Zecpass is software. It is not an exchange, a custodian or an adviser. Read the security page before you rely on it.