TECHNICAL OVERVIEW

RTP Fixed-Price Launch

A structured raise on Uniswap v4: one price for every participant, locked liquidity, full refunds on failure and a pool fee split frozen when each launch is created, before any contribution is accepted.

Document	RTP Fixed-Price Launch: Technical Overview
Version	1.0-draft.26
Date	November 2026
Status	Security review in progress
Code reference	Fixed-Price Launch contracts r3, commit f761cbe
Deployment	Not yet deployed; Robinhood Chain and Base mainnet after internal review

CONTENTS

About this document

This technical overview explains RTP Fixed-Price Launch in plain, verifiable terms: what the contracts do, what they cannot do, and which rules a creator, participant, platform or referrer can rely on.

Abstract

RTP Fixed-Price Launch runs a raise in up to six batches at one price for every participant. If the non-team contributions reach the soft cap, anyone can settle: one transaction creates the token, opens its own Uniswap v4 pool with at least 70% of the raise at the sale price or at a premium fixed in advance, burns the unsold supply and registers the creator's and the team's vesting. If the soft cap is missed, or no one settles in time, every contributor pulls back the full amount. A closed start and a decaying buy tax protect the opening, and the pool fee is split by terms frozen when the launch is created. This paper describes the contracts at version r3.

01 Executive summary

02 Design principles and supported chains

03 Architecture and contract set

04 Whitelist batches and allocation

05 Creator-defined parameters

06 Launch lifecycle

07 Settlement and flow of funds

08 Tokenomics examples

09 Pool fee split and referrals

10 Vesting and the airdrop reserve

11 After settlement: releases and creator yield

12 Opening protection

13 Security model

14 Trust assumptions and known limitations

15 Status, deployment and roadmap

16 Appendix: notation and glossary

17 Appendix: references and revision history

How to read the figures

Bounds quoted in this document are hard limits enforced by the contracts, unless a value is marked as a deployment setting. Fee figures are the caps compiled into the contracts and the terms of the RTP.fun platform registration. Charts labelled Illustrative use example inputs to show how the formulas behave; they are not projections for any project.

Status at the time of writing

Development and internal testing are complete, and the deployment has been rehearsed on forks of live networks. The contracts are not yet deployed; the initial mainnet deployments, on Robinhood Chain and Base, are made directly, without a public testnet stage. Security review in progress: internal review rounds are under way and an external audit is being arranged. Mainnet deployment follows the review rounds and one complete verification run; the external audit may not be complete by then, and its results are published when it is.

Related papers

Three papers are published together on rtp.sh: *RTP Protocol: Overview*; *RTP Curve Launch: Technical Overview*; *RTP Fixed-Price Launch: Technical Overview*. Where two papers describe the same rule, the technical overview of that launch path is the reference.

Not an offer. Nothing in this document is an offer to sell, a solicitation to buy, or investment, legal or tax advice. Participation in any token raise carries the risk of total loss.

EXECUTIVE SUMMARY

A raise whose rules are code, not promises

RTP Fixed-Price Launch lets a project raise ETH at one fixed price, open its own Uniswap v4 pool at that price, or at a premium the creator fixes in advance, and lock at least 70% of the raise as permanent liquidity. Every term is fixed when the launch is created, and every payout is pulled by its owner from a contract ledger.

For participants

Up to six batches, each Merkle-whitelisted or public, one price for everyone including the team, per-address caps enforced in each batch, no early withdrawal that could fake demand, and a 100% refund pulled by each contributor if the soft cap is not met or no one settles within 72 hours of the end. Raised ETH stays in the contract for the whole raise.

For creators and platforms

At settlement at least 70% of the raise becomes permanently locked liquidity; the creator portion is released linearly on-chain from market opening. After opening, every pool swap pays a 1% pool fee split between the protocol, the integrating platform, referrers and the creator, with the creator share never below 50%.

At a glance

70%
at least this share of every successful raise goes into permanently locked Uniswap v4 liquidity

200 ETH
upper safety bound on the creator-set hard cap

6 batches
maximum per launch, each with its own window, root and caps

100%
refund, pulled by each contributor, if the soft cap is not reached or no one settles in time

50%
of the pool fee always goes to the creator, a floor compiled into the contracts

10%
protocol share of the pool fee, fixed in code; no setting can change it

300 s
closed market after the pool is created, then a 600 s decaying buy tax

0 proxies
no upgrade path; a defect is fixed only by a new deployment

What participants can rely on

- Terms are snapshotted at creation. Later configuration changes apply to new launches only.
- Raised ETH is not lent, staked or invested during the raise, so a failed raise refunds exactly what was paid.
- The pool opens at the sale price, unless the creator set an opening premium of up to 50% at creation; the premium is part of the published terms.
- The platform cannot pause contributions, refunds, claims, trading or settlement, cannot move funds credited to others and cannot upgrade contracts.

DESIGN PRINCIPLES

Neutral infrastructure with visible rules

Token raises fail participants in familiar ways: liquidity that can be pulled, insiders who buy at a different price, parameters that change mid-sale, and refunds that depend on goodwill. The Fixed-Price Launch contracts are designed so that each of these is either impossible in code or disclosed up front.

Neutral by construction

Any front end can register a platform and receive the platform share of the pool fee. RTP.fun is one integrator among potentially many. The contracts contain no logic specific to any single token or community.

Terms fixed at creation

Every launch snapshots its parameters and fee split when it is created. The operator can change the success fee and the team limit for future launches only, within compiled bounds and after a 48-hour public delay. A platform's change to its own fee terms reaches only launches created afterwards.

Liquidity that stays

The pool is seeded by an immutable graduator that holds one full-range position and has no withdrawal path. The rule that matters most to holders is enforced by the absence of code.

Pull, never push

Fees, refunds, vested amounts and claims are credited to on-chain ledgers and collected by their owners, which removes whole classes of re-entrancy and denial-of-service risk.

Immutable contracts

No proxy, no upgrade path and no self-destruct; the only delegatecalls go to two libraries whose addresses are fixed when the core is built. The trust surface is smaller; the trade-off, accepted deliberately, is that there is no emergency stop.

Threat model

The project keeps a threat model, to be published with the code, that quantifies the economic limits of its own design, including the costs of sniping, self-trading and multi-wallet participation.

Where it runs

Question	Answer
Supported chains	Chains whose native coin is ETH and that have a Uniswap v4 PoolManager. The initial mainnet deployments are on Robinhood Chain and Base, made directly without a public testnet stage.
Contribution asset	Native ETH. FundZap lets a contributor pay with another token, which it swaps to ETH before contributing. The core only ever counts ETH, and forced ETH is never counted.
Price discovery	One fixed price for all participants; the pool opens at that price, or at the creator's premium of up to 50% fixed at creation.
Who can create	Any address; display on RTP.fun is a separate platform decision.
Failure path	Full refunds, pulled by each contributor.
Upgradeability	None.

Architecture

A compact, self-contained set of contracts. One core holds the ledger of every launch; settlement creates the token's own Uniswap v4 pool, and dedicated contracts handle platform terms, referrals, token contributions and creator yield.

Separate from Curve Launch

The Fixed-Price Launch contracts deploy their own core, hook, graduator, token, vesting and distributor. They share design patterns with RTP Curve Launch but none of its deployed instances, so neither product can affect the other.

Design stance

Same price for every participant, including the team. The pool opens at the sale price or at a premium fixed at creation. No early withdrawal during the raise. ETH only, on chains whose native coin is ETH and that have Uniswap v4 deployed. Every distribution flows from a verified contract, never from an operator wallet.

Contract set

Contract	Role
FundCore	Registry and ledger of every launch: creation, batch roots, contributions, refunds, claims, permissionless settlement, the fee ledger and the operator's powers. Singleton.
CreateLib · SettleLib	Creation checks and settlement, reached by FundCore through delegatecall to libraries fixed when FundCore is built. They keep FundCore under the size limit and change no permission.
FundGraduator	Initialises the pool and adds full-range liquidity in a single unlock. Takes no graduation fee and has no liquidity-removal path.
FundHook	Uniswap v4 hook: the 1% pool fee and its split between protocol, platform, referrers and creator; the closed-market delay before opening; the decaying opening buy tax.
FundToken	One per launch, minted once at settlement. Fixed supply that can only decrease, through burns by holders: no owner, no mint, no tax, no blacklist, no pause, no proxy.
FundVesting	Per-second linear release of the creator's ETH, team-batch tokens and, if the creator chose it, participant tokens. Schedules are irrevocable.
FundDistributor	Merkle airdrops of launch tokens and each launch's airdrop reserve; recipients claim, or anyone pushes tokens to the listed account.
FundPlatforms	The register of platforms and the fee terms each one sets within the compiled caps.
FundReferral	Each launch's referrer list, what each referrer can draw, and social identities bound to wallets.
FundZap	Contributing with another token: swaps it to ETH through the pool manager and contributes for the caller.
FundVault · ERC4626Adapter	Optional deposit of the creator's undrawn ETH, after settlement, into an enabled ERC-4626 vault over WETH; one adapter per venue.
FundRouter · FundQuoter	Trading and quotes for fixed-price pools, in ETH.

Creator-set, within safety bounds

200 ETH
hard cap: creator-set, up to this bound

50%
soft cap: creator-set, at least 50% of the hard cap and 0.5 ETH

1 to 30 days
raise duration: creator-set, starting at least 24 hours after creation

0 to 50%
opening premium over the sale price: creator-set, fixed at creation

FIXED-PRICE LAUNCH

Up to six batches, one price

A fixed-price launch is divided into one to six consecutive batches. Each batch has its own start time and runs until the next one starts, and has either a Merkle whitelist root or a public flag, an optional batch cap and a per-address maximum. Every batch sells at the same price.

Example batch timeline Illustrative

Batch 0	Batch 1	Batch 2	Batch 3	Batch 4	Batch 5
Team. Vested tokens, excluded from the soft cap	Invited KOLs. Whitelist	Sponsors. Whitelist	Community draw. Reproducible lottery	Community tier. Whitelist	Public. Optional

Windows are contiguous, start-inclusive and end-exclusive, from startTime to endTime. Categories are an off-chain arrangement chosen by the creator; the contracts only see windows, roots, caps and whether a batch is the team batch. A batch's unused capacity passes to the batches after it.

Whitelist integrity

- Each leaf commits to the chain id, the FundCore address, the launch id, the batch id, the account and its cap, and is double-hashed. A proof cannot be replayed across chains, launches or batches.
- The creator writes each batch root once, at least 12 hours before the launch starts, and publishes a hash and URI of the list in the same transaction's event, so anyone can rebuild the tree.
- A batch that has neither a root nor a public flag accepts no contributions.
- Allocation lotteries are run off-chain by the platform. The seed is the hash of a pre-announced Ethereum mainnet block, taken once that block is finalised, whichever chain the launch runs on, and a published fixed-point algorithm lets anyone reproduce the result. The contracts only see the resulting Merkle roots.

Contract-enforced caps

- Per address, within a batch: the lower of the leaf cap and the batch maximum. Limits are counted per batch and are not added up across batches, so an address admitted to several batches can use its limit in each of them.
- A contribution above the address's remaining limit, the batch's room or the hard cap's room is filled up to it, and the excess is returned in the same transaction.
- Minimum contribution 0.001 ETH, checked on the amount sent. Contributions are counted in native ETH only.

The team batch

- Optional; always batch 0; capped at 20% of the hard cap. The 20% is a deployment setting: the operator can change it only for launches created afterwards, after a 48-hour delay, and never above 25%.
- Team contributions do not count toward the soft cap, so a team cannot buy its way to success.
- Team tokens are vested on-chain, and an address that contributed to the team batch cannot contribute to another batch.

Example preparation timeline Illustrative

T − 7 d	T − 7 to T − 2 d	T − 36 h	T − 24 h	T − 12 h	T
Launch created; notice period starts (minimum 24 h)	Warm-up; published contribution metrics accrue	Snapshot: list and weights published with their hash	Draw: seed revealed, winners published	Last moment to write batch roots	Batches open in sequence

Only the 24-hour notice period and the 12-hour root deadline are enforced by contract; the rest of the schedule is an example of how a creator may prepare.

Customisable within bounds

Creators shape their own raise: supply, caps, timing, batches, liquidity share, an optional opening premium, vesting and an optional airdrop reserve. FundCore validates every value against hard bounds at creation and reverts if any is out of range. Once created, nothing can be edited. Where a configuration looks weak, the RTP.fun app shows official risk warnings and reference suggestions; this is product-layer guidance, not an on-chain rule.

Parameter	Creator chooses	Bound enforced by contract
Total supply	Initial supply of the launch token	1 million to 1 trillion tokens (18 decimals); immutable
Hard cap / soft cap	Maximum and minimum raise	Hard cap ≤ 200 ETH; soft cap ≥ 0.5 ETH and ≥ 50% of the hard cap, and reachable without the team batch
Timing	Start and end times	Start ≥ creation + 24 h (notice period); duration 1 to 30 days
Batches	1 to 6 windows; root or public flag; caps	Team batch ≤ 20% of the hard cap (deployment setting, compiled maximum 25%); at least one non-team batch
Liquidity share	Portion of the raise placed in the pool	At least 70%; permanently locked
Opening premium	How far the pool's opening price sits above the sale price	0 to 50%; fixed at creation
Creator ETH vesting	Cliff and linear release	Cliff 0 to 365 days; release 90 to 1,460 days; together at most 1,460 days
Team token vesting	Cliff and linear release	Cliff 30 to 1,460 days; release 90 to 1,460 days; together at most 1,460 days
Airdrop reserve	Optional amount, fixed list, vesting	≤ 5% of supply; team batch cap plus reserve ≤ 20% of the tokens a full raise sells; cliff 30 to 1,460 days; release 90 to 1,460 days; together at most 1,460 days
Participant release	Claim schedule for participant tokens	Default: everything claimable from settlement. Optional staged release: at least 20% at market opening, the rest after a cliff of at most 90 days over at most 365 days
Yield venue	Optional; named at creation	Only a venue adapter the operator has enabled on that chain
Payout address	Beneficiary of vested creator ETH	Non-zero; fixed at creation

Vesting bounds by schedule

Schedule	Cliff	Linear release	Cliff plus release	Starts
Creator ETH	0 to 365 days	90 to 1,460 days	at most 1,460 days	market opening
Team-batch tokens	30 to 1,460 days	90 to 1,460 days	at most 1,460 days	market opening
Airdrop reserve	30 to 1,460 days	90 to 1,460 days	at most 1,460 days	market opening
Participant tokens, if staged	0 to 90 days	up to 365 days	at least 20% free at opening	market opening

What a creator cannot set

- The unit price. It is derived by the contract from the supply, the hard cap and the liquidity share.
- Any opening price other than the sale price plus the premium fixed at creation.
- Any free token allocation other than the bounded, vested airdrop reserve.

What no one can do

- Set a lock or release outside the bounds above. Bounds are exposed as read-only views for front ends.
- Change anything after creation. Parameters, roots and reserve lists have no setter.

A one-way state machine

Every fixed-price launch moves through a small set of states, each written once. Time windows are start-inclusive and end-exclusive, and settlement and refunds share one deadline, so at any second exactly one of them can run.

Step 1. Create	Step 2. Pending	Step 3. Active	Step 4. Evaluate
Parameters validated and snapshotted. A notice period of at least 24 h begins. A platform creation fee, if the platform set one (at most 0.1 ETH), is paid here and is not refunded.	Batch roots written once, at least 12 h before the start, with the list hash and URI published. The creator may cancel before the start.	Batches open in order; caps enforced; no withdrawals during the raise.	Success if non-team contributions reach the soft cap, once the end time passes or the hard cap fills.

Success: settled

Permissionless settlement, from the moment the raise succeeds until 72 hours after the end time. Anyone can call settle. One atomic transaction deploys the token, creates and seeds the pool, burns unsold supply, books the success fee and registers vesting.

Claim and protected opening. Participants can claim from settlement, with no deadline. Trading opens 300 s after pool creation, followed by a 600 s decaying buy tax.

Failure: refunding

The soft cap was not reached by the end time, or no one settled within 72 hours after it. The launch is refunding from that moment, with no approval from anyone.

100% refund, pulled by each contributor. Every contributor withdraws the full amount contributed, to an address of their choice. No token is minted.

Pending to cancelled: only before the start

The creator alone may cancel, and only before the start time, when no contribution can yet exist. Once the raise has started, no one can cancel it.

Settlement cannot be blocked

Settlement never reads contract balances, uses only a pool key bound to FundHook, derives the token address deterministically and runs in constant gas regardless of the number of contributors.

All or nothing

If any step of settlement fails, the whole transaction reverts and the launch stays as it was, so it can be retried before the deadline. Failures are never silently swallowed.

No early exit

Withdrawals during a raise would let allocations be recycled and demand be faked. The exits are a full refund on failure, or a sale on the open market after opening.

Where every unit goes

At settlement, the raise and the token supply are allocated by formula in a single transaction. The allocations below follow directly from the parameters fixed at creation.

ETH raised (R), held in the FundCore ledger

Liquidity, at least 70%	Paired with tokens in one full-range v4 position at the opening price. Permanently locked.
Success fee	R × the success fee, to the protocol treasury. 1% in this deployment; compiled maximum 3%; a change applies only to launches created at least 48 hours later.
Creator, the remainder	Placed in FundVesting on the creator's linear schedule, which starts at market opening.

Token supply (ts): FundToken minted once

Participants	R × rate, team batch included, claimed by each contributor.
Pool	Liquidity ETH × rate, reduced by the opening premium if one was set.
Team batch	Into FundVesting, pulled by each member.
Unsold	Burned at settlement, together with any tokens the pool does not take.
Airdrop reserve	Optional; minted on top of ts and sent to FundDistributor; released from market opening.

The creator never receives the raise in a lump sum. Even with the minimum schedule (no cliff, 90-day release) the creator portion is released evenly over 90 days from market opening. Raised funds are not invested during the raise and are refunded in full on failure.

Pricing identity

```
rate = floor( ts × 10,000 / ( hardCap × (10,000 + lpBps) ) )
```

Here ts is the total supply in token base units, hardCap is the hard cap in wei, lpBps is the liquidity share in basis points (at least 7,000), and rate is token base units per wei, which is the same as tokens per ETH. A full raise at the hard cap exactly uses the supply sold to contributors, the team batch included, and the supply placed in the pool. The pool receives

```
poolTokens = liquidityETH × rate × 10,000 / (10,000 + premiumBps)
```

With no premium the pool opens at the sale price, so there is no discount gap at opening for participants to sell into. With a premium the pool opens that much higher, by at most 50%, and the tokens the pool does not take are burned with the unsold supply.

Opening price
equals the sale price, plus the premium if one was set at creation

No graduation fee
FundGraduator takes nothing from the liquidity portion

Constant gas
settlement cost does not grow with the number of contributors

Supply after settlement

Total supply equals tokens sold plus tokens in the pool plus any reserve minted. Unsold tokens and rounding dust are destroyed with a true burn rather than sent to a dead address, so on-chain supply reflects reality and no oversized holder appears on wallet-cluster maps. Distribution always flows from verified contracts (FundCore, FundVesting, FundDistributor), never through an operator wallet.

How the formulas allocate supply Illustrative

Example inputs, chosen only to demonstrate the mechanics: supply 1,000,000,000 tokens, hard cap 100 ETH, soft cap 50 ETH, liquidity share 70%, no opening premium, success fee 1%, a team batch capped at 10 ETH and fully used, and an airdrop reserve of 20,000,000 tokens (2% of supply). The derived rate is 5,882,352 tokens per ETH (rounded down).

Example A. Full raise of 100 ETH

Allocation	Tokens	Share
Public and whitelist participants	529.4M	51.9%
Liquidity pool (locked)	411.8M	40.4%
Team batch (vested)	58.8M	5.8%
Airdrop reserve (vested)	20.0M	2.0%
Final supply	**1,020M**	

No unsold tokens to burn beyond rounding dust. Final supply is 1,000M plus the 20M reserve. ETH: 70 to liquidity, 1 to the protocol treasury, 29 to the creator.

Example B. Raise of 60 ETH, soft cap met

Allocation	Tokens	Share
Public and whitelist participants	294.1M	47.4%
Liquidity pool (locked)	247.1M	39.8%
Team batch (vested)	58.8M	9.5%
Airdrop reserve (vested)	20.0M	3.2%
Final supply	**620M**	

400.0M unsold tokens are burned at settlement. The 50 ETH of non-team contributions meets the soft cap. ETH: 42 to liquidity, 0.6 to the protocol treasury, 17.4 to the creator.

In both examples the price paid, the sale price and the pool opening price are identical. A partial raise does not lower the price; it reduces supply through the burn, preserving the liquidity-to-supply relationship of the design.

Creator-controlled share

The limit on team batch plus reserve is measured against a full raise. In Example A the team batch (58.8M) plus the reserve (20.0M) equals 13.4% of the tokens a full raise sells, within the 20% limit. In Example B, a partial raise, the same 78.8M is 22.3% of the tokens actually sold (352.9M) and 12.7% of the final supply, because the team batch and the reserve do not shrink with the raise. Front ends must display this combined figure for the actual raise, and the reserve is labelled a creator-designated free allocation.

Example values are chosen for readability only. They are not recommendations: allocation splits are fully creator-defined within the contract bounds, and the RTP.fun app adds risk warnings and reference suggestions for weak configurations.

POOL FEE SPLIT

How the pool fee is shared

After opening, every swap in a fixed-price pool pays a 1% pool fee through FundHook. The hook splits that fee in basis points of 10,000 and credits each share to a pull-based ledger. Each launch freezes the terms of its platform when it is created.

Split of the pool fee on the RTP.fun platform



Per-trade referral 5%, list referral 10%. Opening-tax proceeds are accounted separately and are never part of this split.

Share	RTP.fun terms	Rule enforced by contract
Protocol	1,000 bps (10%)	Fixed in code, taken before the other shares and rounded up. No setting can lower, raise or redirect it.
Platform	2,000 bps (20%)	Set by each platform in its own registration, capped at 2,000 bps; frozen per launch at creation.
Referrals, combined	1,500 bps (15%)	Capped at 2,000 bps. Split between a per-trade referrer (500 bps on RTP.fun) and the launch's list (1,000 bps).
Creator	The remainder (55%)	Never below 5,000 bps, whatever the other settings; takes the rounding dust as well.

Per-trade referral

A swap may name a referrer. The per-trade share stays available after trading opens, so integrators and communities are credited for the volume they bring.

List referral pool

Credited on every swap and shared among up to five referrers, wallets or social identities, named at creation in fixed parts. The list cannot change. A social identity's part waits until the identity is bound to a wallet.

No fee is lost

If a swap names no valid per-trade referrer, that share joins the list pool; if the launch has no list, it goes to the creator's payee. Rounding never leaves a share unassigned.

Other platform fees. A platform may also set a creation fee, at most 0.1 ETH, and a fee on contributions made through FundZap, at most 1%. Each launch freezes these with the rest of its platform's terms, and the app shows them before anyone signs.

Releases you can verify on-chain

Example release schedules Illustrative

Schedule	Example parameters	Share released at 90 d	180 d	365 d	720 d
Creator ETH	no cliff, 365 d release	24.7%	49.3%	100%	100%
Team tokens	180 d cliff, 540 d release	0%	0%	34.3%	100%
Airdrop reserve	30 d cliff, 90 d release	66.7%	100%	100%	100%

Days after market opening. All schedules start at market opening, 300 s after settlement, and release linearly per second after their cliff. Absolute cliff-end and end timestamps are emitted as events at settlement.

Vesting rules

- Schedules are registered at settlement from the snapshot taken at creation and are irrevocable. No one, the platform included, has a function to withdraw or alter them.
- Team members pull their own share pro rata to their team-batch contribution. Totals reconcile exactly at the end of the schedule.
- Participant tokens can by default be claimed in full from settlement. A creator may instead set a staged release through FundVesting at creation: at least 20% at market opening, the rest after a cliff of at most 90 days, over at most 365 days.
- Front ends must display the complete timetable, including cliff-end and end dates, before anyone contributes.

Note. Release speed depends on the release period, not the cliff. At the minimum setting the creator portion can be fully released within 90 days of market opening.

Optional airdrop reserve

- Any creator may declare a reserve of at most 5% of supply, with the team batch cap plus the reserve limited to 20% of the tokens a full raise sells. In a partial raise the combined share of the tokens actually sold is higher; see the tokenomics examples.
- The recipient list is supplied and signed by the creator, fixed at creation through its root, hash and URI, and cannot be amended or extended.
- Minted only if the launch settles successfully. A failed launch mints nothing.
- A cliff of at least 30 days and a linear release of at least 90 days, both from market opening, together at most 1,460 days.
- Activated by anyone after settlement. It expires 365 days after the end of its release, or 365 days after activation if that is later. After expiry anyone can trigger the burn of what was not claimed; it is never returned to anyone.
- Delivered by FundDistributor: recipients claim, or anyone pushes tokens to the account in the list. A push cannot change the list, the amounts or the recipients.

Cliffs and linear release spread selling over time and give the market time to inspect the list. They are not a price guarantee.

AFTER SETTLEMENT

After settlement: releases and creator yield

Settlement turns the raise into locked liquidity, a creator portion on a linear schedule and participant tokens. Two rules keep this simple to verify: no yield during the raise, and yield only on the creator's own ETH afterwards.

During the raise

- Contributions stay in FundCore. They are not lent, staked or deposited anywhere.
- If the soft cap is not reached, or settlement does not happen within 72 hours of the end, every contributor pulls back 100% of the amount paid.
- No one, the platform included, can move raised ETH before settlement.

After settlement

- The creator portion is released per second from market opening on the schedule fixed at creation. There is no lump-sum payout and no early-release option.
- Participant tokens can be claimed in full from settlement, unless the creator set a staged release at creation.
- Liquidity is permanently locked and is never used for yield.

Optional yield on the creator's ETH

A creator may name a yield venue when creating the launch. Only a venue adapter the operator has enabled on that chain can be named, and the choice is then one of the launch's terms. After settlement the creator's payout address may move the creator ETH not yet withdrawn from FundVesting into that venue, an ERC-4626 vault over WETH. Amounts unlock on the same timetable whether or not yield is active.

- Applies only to the creator's own ETH, after settlement. Participant funds, refunds and pool liquidity are never deposited.
- Venue adapters are enabled per chain after a 48-hour delay, each with a fixed capacity, and can be disabled at once, which stops new deposits only. Candidate venues have been tested on forks of Ethereum, Base and Arbitrum. None is enabled yet, and on a chain with no enabled venue the option is off.
- The vault charges no fee and has no fee setting.

Risk note. Lending protocols carry smart-contract, liquidity and market risk. Yield is variable and not guaranteed, and an opted-in creator bears the risk that a deposit returns less than expected. Participants are unaffected.

No raise-period yield
raised ETH is held, not invested

Same schedule
yield never accelerates creator release

No vault fee
the vault takes nothing from creator yield

A closed start, then a decaying buy tax

A fixed-price pool opens at a known time and a known price, which invites an opening rush. FundHook therefore carries a two-part protection, enforced inside the hook so that it applies to every router and every direct caller of the pool.

Seconds after settlement	0 to 300	300	450	600	750	900 and later
Additional tax on buys	trading closed	50%	37.5%	25%	12.5%	0%

openAt = settlement + 300 s. The 1% pool fee applies on top in every row. Parameters are immutable in FundHook and shared by every fixed-price launch of the same deployment.

Phase 1. 300 s closed

When settlement creates the pool, the hook records an opening time 300 seconds later. Until then every swap reverts, whatever the route. Participants use this window to claim, and front ends open in unison.

Phase 2. 600 s decay

From opening, buys pay an additional tax that starts at 50% and falls linearly to zero over 600 seconds. It applies to exact-input and exact-output buys alike. Sells pay only the 1% pool fee.

Afterwards

The pool trades with the 1% pool fee only. Opening-tax proceeds go to the protocol treasury in full and are never shared with the creator, the platform or referrers, so self-dealing cannot recover them.

Why it is built this way

- A Uniswap v4 swap only requires an unlocked pool manager, so any contract can trade a pool directly. A router-level tax could be bypassed; a hook-level rule cannot.
- The pool is initialised and fully seeded in the same unlock. An empty pool awaiting liquidity could have its price pushed at no cost, so that pattern is explicitly prohibited.
- With no premium the opening price equals the sale price, so the tax targets the opening rush rather than an artificial price gap.

Design note. Once claimed, tokens are freely transferable, and someone could create a separate pool before the official opening. This does not affect the official pool's price or tax schedule, and it is documented in the threat model.

What is enforced, and by what

Trust claims are only as good as the mechanism behind them. This section separates guarantees enforced by contract code from commitments made at the platform layer.

Enforced by contract

- No upgrade path: no proxy, no upgradeable delegatecall, no self-destruct, no initialiser to front-run.
- Pool liquidity has no removal path; only the graduator adds it.
- The treasury is fixed as an immutable constructor parameter of FundCore, FundHook and FundGraduator.
- Parameters and fee split snapshotted per launch; no setter for existing terms, roots or reserve lists.
- Pull payments behind re-entrancy guards; ETH only for contributions; forced ETH is never counted.
- Settlement is atomic, permissionless and constant-gas; refunds need no approval.

Powers, in full

Who	What they can do
Operator (the platform Safe)	Pause the creation of new fixed-price launches. Change the success fee (at most 3%) and the team limit (at most 25%), each after a 48-hour delay and only for launches created afterwards. Enable yield-venue adapters after a 48-hour delay, with a fixed capacity, and disable them at once, which stops new deposits only. Propose a new referral attestor, which takes 7 days, or revoke it at once; cancel an identity binding during its 72-hour public notice (the treasury can cancel too).
Each platform	Set its own fee terms and payee within the compiled caps: platform share up to 20% and referral budget up to 20% of the pool fee, a creation fee up to 0.1 ETH and a FundZap fee up to 1%. A change reaches only launches created afterwards; the payee can be moved at any time.
Never, by anyone	Pause trading, contributions, refunds, claims or settlement; move funds credited to others; touch a pool; change an existing launch; change the treasury; upgrade a contract.

Platform-layer controls

Display on RTP.fun

Launch creation is permissionless, so look-alike launches can exist on-chain. RTP.fun shows as verified only launches that pass name, symbol and logo checks; others are labelled unverified, and verification is never inferred from on-chain fields.

Terms and regions

On RTP.fun, users and creators sign the applicable terms with a gas-free wallet signature before creating, contributing or trading, and participation is not offered in restricted regions. These controls do not bind someone who calls the contracts directly. Refunds and claims are never blocked.

Airdrop list rules

Lists come from the creator and are signed by the creator. Screening can only remove addresses, never add them. Eligibility may not depend on buying, holding or promotion, and snapshots precede announcements.

Design boundaries

- On-chain rules cannot identify multiple wallets controlled by one person. A team using secondary wallets in public batches is addressed by platform-layer controls and documented in the threat model.
- Platform-layer controls such as terms signatures apply on RTP.fun; they do not cover a user who calls the contracts directly or uses another front end.
- Sequencer behaviour on each network is outside the contracts' control. The design assumes worst-case ordering, and the failure direction of a stalled settlement is a full refund.

Trust assumptions and known limitations

These are the places where the contracts depend on something outside themselves, or where a protection has a stated limit. The threat model treats each of them in detail.

Topic	What holds, and what does not
Several wallets of one person	On-chain rules cannot identify them. A team member can use another wallet in a public batch; front ends must say so.
Limits per batch	Per-address limits are counted in each batch and not across batches.
Team share in a partial raise	The limit on team batch plus reserve is measured against a full raise; in a partial raise their share of what was sold is higher.
One party, several roles	The same party can be creator, platform and referrer and take all of the pool fee except the protocol's share. That is allowed; the protocol's share holds.
Platform payee	A platform can move its payee at any time; fee numbers are frozen per launch, the payee is not.
Identity bindings	They depend on the attestor key and on someone watching the 72-hour notice; the operator can cancel a binding in that window and revoke the attestor.
Yield venues	A venue can lose value or refuse a redemption; that risk is the opted-in creator's alone. Deposits and redemptions take no minimum, so a venue's depth is checked before it is enabled.
Pools made by others	Claimed tokens can be placed in another pool before the official opening. The official pool's price and schedule are not affected.
Sequencer ordering	Outside the contracts' control on every network; a stalled settlement fails towards a full refund.
Software risk	Smart contracts may contain undiscovered defects. Internal review rounds are under way and an external audit is being arranged; it may not be complete at mainnet deployment.

From testnet to mainnet

Where the protocol stands

- Development and internal testing are complete: settlement, pricing identities, the fee split, opening protection and cross-chain deployment are covered by unit, invariant, fuzz and fork tests.
- Every release is gated by its full test suite, mutation testing and static analysis.
- Security review in progress: internal review rounds are under way and an external audit is being arranged. Results are published when each is complete.

Deployment in three batches

- Deployment is three Safe transactions from a dedicated deployment Safe, each below half of the 16,777,216 gas (2^{24}) that one transaction may use.
- Before each batch is signed, a precheck reads the chain to confirm that the earlier batches are in place. The core is deployed last and starts with creation paused.
- A fourth transaction, from the platform Safe, opens creation once the deployment has been verified on-chain.
- The full sequence has been rehearsed through a Safe on local forks of Ethereum, Base and GIWA Sepolia.

Roadmap

Stages are sequential milestones rather than calendar commitments, so no dates are given.

Complete	Complete	In progress	Next	Being arranged
Protocol design Parameters, fee split, state machine and threat model	**Development and testing** Full test set, invariants and fork rehearsals	**Internal review and verification** Review rounds, then one complete verification run	**Robinhood Chain and Base mainnet** Deployed directly, without a public testnet stage	**External audit** Results published when complete

Same address on every supported chain. Fixed-Price Launch contracts are deployed through CREATE3 from a dedicated deployment Safe, so they share one address on each supported chain. Because the same address does not prove the same code, each deployment publishes a manifest of addresses, runtime and init code hashes, configuration hashes and the state to read after deployment, for independent verification, in a public deployment repository that holds no source code.

APPENDIX

Notation and glossary

Notation

Symbol	Meaning
ts	Total supply in token base units (18 decimals)
hardCap, softCap	Maximum and minimum raise, in wei
R	ETH raised, team batch included
lpBps	Liquidity share of the raise, in basis points, at least 7,000
premiumBps	Opening premium over the sale price, in basis points, 0 to 5,000
rate	Token base units per wei, which equals tokens per ETH
openAt	Market opening: settlement + 300 seconds

Glossary

Term	Meaning
Basis points (bps)	Hundredths of a percent. The pool fee split is expressed in parts of 10,000.
Fee split	The shares of the pool fee credited to the protocol, the platform, referrers and the creator, frozen per launch at creation.
Hook	A Uniswap v4 contract invoked by the pool on swaps and liquidity events.
Locked liquidity	A pool position held by a contract with no withdrawal function. The liquidity exists and is tradable, but no one can remove it.
Soft cap / hard cap	The minimum raise for a fixed-price launch to succeed, and the maximum it can accept.
Batch	A time window within a fixed-price launch, with its own Merkle whitelist or public flag, an optional cap and a per-address limit.
Merkle root	A single hash committing to a whole whitelist or airdrop list, against which each participant proves inclusion.
Settlement	The single, permissionless transaction that finalises a successful fixed-price launch: token deployment, pool creation, burns and vesting.
Market opening	The moment trading opens in a fixed-price pool, 300 seconds after settlement. Every vesting schedule starts here.
Opening premium	An optional amount, 0 to 50%, by which the pool's opening price exceeds the sale price; fixed at creation.
Success fee	A share of the raise booked to the protocol treasury at settlement: 1% in this deployment, at most 3%.
Cliff / linear release	A period with no release, followed by per-second proportional release until the full amount is available.
Pull payment	A design in which recipients withdraw amounts credited to them, instead of contracts sending funds during other operations.
Creator-designated free allocation	The front-end label for the optional airdrop reserve, the only free token allocation a fixed-price launch can contain.

APPENDIX

References and revision history

References

Key	Reference
[UNI-V4]	Uniswap Labs, Uniswap v4 Core: the singleton PoolManager and hooks.
[EIP-7825]	Transaction Gas Limit Cap: at most 16,777,216 (2^{24}) gas per transaction.
[ERC-20]	Token Standard.
[EIP-712] [ERC-1271]	Typed structured data signing; signature validation for contracts.
[CREATEX]	CreateX factory, CREATE3 deployment to the same address on every chain.
[BUSL]	Business Source License 1.1.
[ROBINHOOD-CHAIN]	Robinhood Chain, an Ethereum Layer 2; mainnet chain id 4663, testnet 46630.
[ERC-4626]	Tokenized Vault Standard, used by the optional yield venues.
[SAFE]	Safe 1.4.1 (SafeL2) and the Safe Transaction Builder format, used for the deployment batches.
[BASE]	Base, an Ethereum Layer 2 on the OP Stack; mainnet chain id 8453.
[CODE-FP]	RTP Fixed-Price Launch contracts, r3, commit f761cbe. The source is published with the public release.

Revision history

Version	Date	Change
draft 18	2026-11-03	Initial draft, in two documents.
draft 19	2026-11-04	Every statement checked against the contracts and corrected where it differed.
draft 20	2026-11-04	Names: RTP Protocol, RTP Curve Launch, RTP Fixed-Price Launch. Material not yet public is no longer described as published.
1.0-draft.21	2026-11-04	Three documents: the protocol overview and a technical overview of each launch path. Document control, notation, trust assumptions and known limitations, references.
1.0-draft.22	2026-11-04	Security status stated as it stands: internal review rounds under way, external audit being arranged. Deployment plan: GIWA stays on testnet; both launch paths go to Robinhood Chain mainnet after the review rounds and one complete verification run.
1.0-draft.23	2026-11-05	Document identifiers removed; the papers refer to each other by title. Runtime codehashes and the deployment manifest are published in the code repository; rtp.sh lists the addresses.
1.0-draft.24	2026-11-05	References name GIWA and Robinhood Chain. The Fixed-Price Launch targets Robinhood Chain mainnet directly.
1.0-draft.25	2026-11-05	Fixed-Price Launch also targets Base mainnet. Deployment manifests go to a public repository that holds no source code. RTP.fun named for the platform layer; the Curve Launch fee hook stated as CREATE2.
1.0-draft.26	2026-11-05	Fixed-Price Launch: the success fee in this deployment is 1%; the worked examples show it.

Licensing

The Fixed-Price Launch contracts and libraries, FundToken and FundZap included, are licensed under the Business Source License 1.1 (Licensor RTP.fun, change date 2029-10-25, then MIT). FundRouter, FundQuoter and the interfaces are MIT. The licences grant no trademark rights.

Verification and contact

Only addresses listed in the official deployment registry on rtp.sh are official, whatever name a front end presents. Security reports: security@rtp.fun, by private disclosure. Licensing enquiries: licensing@rtp.fun.

Important information. This document describes software under development and testnet deployments that carry no real value. Parameters, designs and timelines may change before any mainnet deployment, and a mainnet deployment may come before the external audit is complete. Digital assets are volatile and may lose all value. Smart contracts may contain undiscovered defects despite review. Availability may be restricted in certain jurisdictions. This document is not an offer, a solicitation or advice of any kind.