

URM Stablecoin

Symbol: \$URM | v1.7 | June 7, 2026

Authors: Crypto Surfer & Hedgex

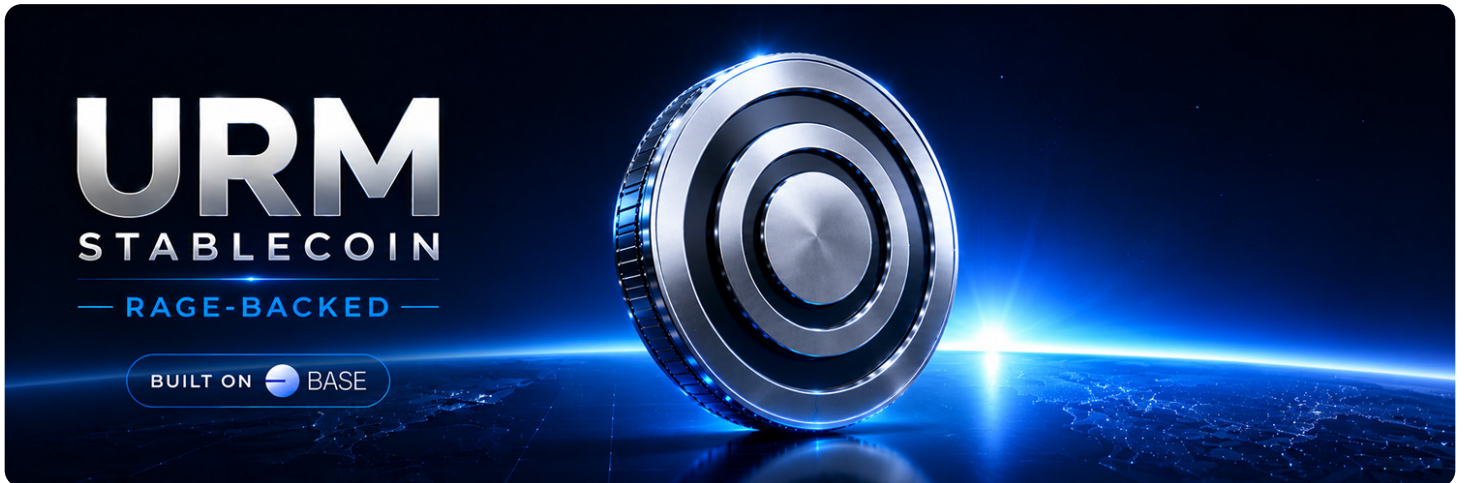


Table of Contents

1. Overview
2. The Fortress
 - URM/USDC pool
 - Towers
 - Supply
3. Core Mechanics
 - Backing capacity
 - Peg defense
4. setConfigs

1. Overview

URM is a stablecoin soft-pegged to USDC, deployed on Base. A single contract — the Fortress — owns the URM/USDC liquidity, runs automated peg defense, mints and burns URM supply, and routes protocol revenue to

registered contracts ("towers") as real yield.

URM is backed by the USDC in the Fortress's LP position, plus a defense leg of RAGE and underlying assets. The RAGE sits in RageDepot and the ecosystem RAGE/USDC pool; the underlying — HESTIA and CIRCLE — is held by the Fortress directly.

Governance splits two ways: a two-owner process registers or removes towers and can freeze the set permanently, while a single automator tunes parameters and drives peg defense.

2. The Fortress

The Fortress is the protocol-owned contract that controls URM peg dynamics.

- It owns the URM/USDC LP position (one Uniswap V3 NFT) and increases it as the protocol grows.
- It runs an automator-driven keeper that performs peg defense, with a 3-minute minimum between defense actions.
- It mints URM to itself within the Backing capacity and burns its own excess balance.
- It collects pool fees, converts them to URM, and distributes them to towers as real yield.
- It supports up to 9 tower contracts, each approved through a two-owner process.

Two-owner actions require both RCE owners to submit the same call: registering or removing a tower, freezing the tower set (`setFortressFinal`), and pointing the URM token at the Fortress. Either owner alone can lock the LP NFT into the Fortress, once. Automator actions cover `setConfigs`, `setTowersConfigs`, `pegDefense`, and sweeping stray non-protocol tokens. URM, HESTIA, and CIRCLE can never be withdrawn.

URM/USDC pool

URM trades against USDC in a single Uniswap V3 position on the 0.05% fee tier. Fortress liquidity is concentrated in a tight \$0.99–\$1.01 band around peg, so slippage stays well inside the fee. `collectFees` claims the position's accrued fees and converts them to URM for distribution to towers as yield. The band fully contains the defense thresholds, so every peg-defense swap lands inside the position's range.

Towers

A tower is a registered contract that participates in URM backing — a CDP, a yield venue, or any contract that holds URM and USDC on the protocol's behalf. The Fortress holds up to 9 tower slots; each is added or removed only when both owners submit the same address. Once `setFortressFinal` is confirmed, the set is frozen permanently.

Each tower carries a status:

- **Disabled (0)**. Ignored entirely.
- **Backing only (1)**. Its balances count in the Backing capacity, but it receives no URM.
- **Enabled (2)**. Counted in the Backing capacity and eligible to draw URM from the Fortress during sync.

Each tower reports five figures through `fortressData`: USDC pooled, USDC stored, URM pooled, URM stored, and URM borrowed. Pooled and stored USDC add to backing capacity; all reported URM is excluded from the supply the Fortress must back — towers are treated as self-collateralised. Enabled towers can pull up to 25% of the Fortress's URM balance per day, drawn from the Fortress's minted reserve rather than freshly minted on demand. Each tower also carries a `recipientPercent`, its share of distributed yield.

Yield reaches towers as URM and is real, not emitted. It is sourced from URM/USDC pool fees and from tower profit returned through `payFortress`; the Fortress splits the total across towers by `recipientPercent` and keeps any unallocated share as its own yield.

Supply

The Fortress mints only to itself. When its balance falls below `requestSupplyMin` (default 50,000 URM), `pegDefense` calls `requestSupply`, minting `requestSupplyPercent` of supply (default and maximum 10%, once per day). When its balance exceeds `burnSupplyMin` (default 100,000 URM), the excess is burned. The working reserve therefore floats between the two thresholds and feeds tower sync, above-peg sales, and burns.

3. Core Mechanics

Backing capacity

Backing capacity is the ceiling on how much URM the Fortress may issue — total defense capacity minus the URM the Fortress is responsible for backing.

Defense capacity (in USDC) sums:

- USDC in the URM/USDC position, at face.
- USDC pooled and stored by every backing-enabled tower, at face.
- Smaller of: $\text{RageDepot RAGE value} \times \text{assetsValuePercent}$, or $\text{ecosystem RAGE/USDC pool USDC} \times \text{ragePoolPercent}$ — external RAGE capacity, gated by whichever side is shallower.
- HESTIA and CIRCLE held by the Fortress at TWAP \times `assetsValuePercent`.

URM obligation (in URM) is total supply minus:

- URM held in the URM/USDC position.
- URM held by backing-enabled towers.
- URM held by the Fortress itself.

Backing capacity floors at zero, and it is the cap on above-peg defense: a single action can sell no more URM into circulation than the backing capacity available at that moment. The haircuts keep this conservative — USDC counts at face, while RAGE and underlying are discounted by `assetsValuePercent` (default 75%).

Peg defense

The automator calls `pegDefense` with a `defenseSize` in USDC. The Fortress reads the URM/USDC TWAP and acts only outside the deadband; a 3-minute minimum separates defense actions of the same type. Each call may also top up supply, collect fees, run tower sync, and add liquidity before defending.

`defenseSize` is capped dynamically, not by a fixed limit: the ceiling is a quarter of the URM/USDC position's two legs combined, with URM counted at par against its USDC. Because the cap tracks the live position, defense capacity widens as liquidity deepens and tightens as the pool is drawn down — so no single action can move more than a small share of the pool.

Above peg ($URM > \text{abovePegThreshold}$, default \$1.005):

- The Fortress sells roughly `defenseSize` of URM for USDC, capped by the Backing capacity.
- `abovePegBuy%` of the USDC funds the buy leg; the remainder is added to the URM/USDC position.
- The buy leg routes by RAGE price versus FMV (or a forced target): below FMV it buys RAGE into `RageDepot`; at or above FMV it buys HESTIA/CIRCLE, growing RAGE's backing.
- A signed counter increases by the USDC spent on the buy leg.

Below peg has two floors:

- The soft floor ($URM < \text{belowPegThreshold}$, default \$0.999) fires only while the counter holds enough earned surplus — it is funded by prior above-peg activity.
- The hard floor ($URM < \text{belowPegThreshold2}$, default \$0.995) fires unconditionally.
- The Fortress raises `defenseSize` of USDC by selling RAGE from `RageDepot` or HESTIA/CIRCLE (routed by RAGE versus FMV, or a forced target), buys URM on the pool, and decreases the counter by `defenseSize`.
- Bought URM accumulates in the Fortress and is burned down to `burnSupplyMin` each cycle, permanently contracting supply.

4. setConfigs

The Fortress exposes two privileged functions, both automator-only and each on a 1-hour cooldown: `setConfigs` tunes Fortress parameters, and `setTowersConfigs` sets each tower's yield share and status. `defenseSize` is not a config — it is passed per call and bounded dynamically by pool depth.

Parameter	Default	Min	Max	Description
<code>assetsValuePercent</code>	75%	50%	100%	Share of <code>RageDepot</code> and underlying value counted as soft defense.
<code>ragePoolPercent</code>	50%	25%	75%	Share of the ecosystem RAGE/USDC pool USDC counted as soft defense.

abovePegThreshold	\$1.005	\$1.001	—	URM/USDC TWAP above which above-peg defense triggers.
belowPegThreshold	\$0.999	—	\$0.999	Soft floor; defends only while the counter holds surplus.
belowPegThreshold2	\$0.995	—	< belowPegThreshold	Hard floor; defends unconditionally.
abovePegDelay	2 min	1 min	1 day	Min interval between above-peg defense actions.
belowPegDelay	2 min	1 min	1 day	Min interval between below-peg defense actions.
abovePegBuy	50%	0%	100%	Share of above-peg USDC routed to the buy leg; remainder to LP.
pegDefenseTarget	0	0	2	Buy/sell asset preference: 0 routes by RAGE vs FMV, 1 prefers RAGE, 2 prefers underlying.
requestSupplyPercent	10%	1%	10%	Share of supply minted per requestSupply call (URM enforces a 1-day cooldown).
requestSupplyMin	50,000 URM	0 (off)	bal + 50,000	Balance below which pegDefense tops up via requestSupply.
burnSupplyMin	100,000 URM	0 (off)	—	Balance above which pegDefense burns the excess (must exceed requestSupplyMin).
syncTowers	true	—	—	Master switch for tower sync inside pegDefense.

Min interval between

collectFeesDelay	7 days	0 (off)	365 days	collectFees runs inside pegDefense.
liquidityIncrease	true	—	—	If true, pegDefense adds URM/USDC liquidity when holding ≥ 1 USDC.
twap	180	60	3600	TWAP interval (seconds) for peg-defense pricing.
slippage	5%	0%	25%	Max swap slippage (0 = off).

setTowersConfigs takes one entry per slot (1–9):

Parameter	Default	Min	Max	Description
recipientPercent	0	0	100	Share of distributed yield to this tower; all slots must sum to ≤ 100 .
status	0	0	2	0 disabled, 1 backing only, 2 backing + sync.