JELLEO | AUTONOMOUS SOLANA AUDIT

AUDIT REPORT · CYCLE 20260428-193720

Hunt cycle · default.

AUDITOR	Kirill Sakharuk · kirill@jelleo.com
CUSTOMER	default
WINDOW	cycle 20260428-193720
CYCLE	20260428-193720
ENGINE SHA	a946e5508f
WRAPPER SHA	4c22024dd3
GENERATED	2026-05-08T22:32:44+00:00

0	0	0	0	0
CRITICAL	HIGH	MEDIUM	LOW	INFO

CONFIRMED · DISCLOSED · FIXED · VERIFIED

12 REJECTED (FALSE POSITIVE)

SIGNED · ED25519

MCowBQYDK2VwAyEAvCFSLBecPuNClei48PWjHuelHlBX9uYZo4wELbQ7b+k=

verify with `audit-pipeline sign verify <file> <file>.sig --pubkey jelleo.ed25519.pub`
public key at
https://jelleo.com/keys/jelleo.ed25519.pub

PLATFORM · V0.1

JELLEO · The underwriting layer for Solana DeFi.

Methodology	jelleo.com/methodology.html
Disclosure	jelleo.com/security.html
Source	github.com/Copenhagen0x/audit-pipeline-cli

Apache-2.0 · contact security@jelleo.com

default · hunt cycle

20260428-193720 · started 2026-04-28T19:37:20+00:00 · engine a946e5508f · wrapper 4c22024dd3

01 — CYCLE SUMMARY

CRITICAL	HIGH	MEDIUM	CONFIRMED
0	0	0	0

TOTAL VERDICTS
12

Critical 0 · High 0 · Medium 0 · Low 0 · Info 12

02 — FINDINGS

SEVERITY	HYPOTHESIS	TITLE	VERDICT	STATUS	POC
INFO	H1-residual-conservation	The post-haircut residual cash on a market (vault - cash_locked_in_orderbook - claimable_pnl - insurance_counter) is con	UNKNOWN / UNKNOWN	REJECTED	—
INFO	H10-init-state-invariants	The post-init state of a market (vault, c_tot, insurance_fund.balance, pnl_pos_tot, pnl_matured_pos_tot, all OI counters	UNKNOWN / UNKNOWN	REJECTED	—
INFO	H11-account-gc-state-leak	free_slot / reclaim_empty_account zeroes every byte of the account slot, so a subsequent materialize_at on the same inde	UNKNOWN / UNKNOWN	REJECTED	—
INFO	H12-arithmetic-bounds-on-mature	The lazy mark-to-market PnL computation (pnl_delta = abs_basis * (K_now - K_snap) / (a_basis * POS_SCALE)) cannot overfl	UNKNOWN / UNKNOWN	REJECTED	—
INFO	H2-haircut-direction	The haircut (positive-PnL claim cap) only ever shrinks claimable PnL, never increases the residual cash that other claim	UNKNOWN / UNKNOWN	REJECTED	—

SEVERITY	HYPOTHESIS	TITLE	VERDICT	STATUS	POC
INFO	`H3-self-trade-cash-flow`	A self-trade (same authority on both sides of a fill) is cash-flow neutral up to fees + IM transitions.	UNKNOWN / UNKNOWN	REJECTED	—
INFO	`H4-vault-balance-conservation`	For every market state transition, the change in vault balance equals the sum of (cash deposited into orderbook + claima	UNKNOWN / UNKNOWN	REJECTED	—
INFO	`H5-permissionless-trigger-surface`	Every public/permissionless instruction that reaches use_insurance_buffer requires either an admin signer OR cannot drai	UNKNOWN / UNKNOWN	REJECTED	—
INFO	`H6-funding-rate-mark-bias`	The funding rate captured by every instruction is computed BEFORE any mark_ewma_e6 / last_effective_price_e6 mutation in	UNKNOWN / UNKNOWN	REJECTED	—
INFO	`H7-catchup-touch-account-pairing`	Every public instruction that advances the engine's market clock (accrue_market_to / accrue_market_to_chunked) is paired	UNKNOWN / UNKNOWN	REJECTED	—
INFO	`H8-keeper-crank-cursor-consumption`	The keeper crank's price-move consumption budget is not reset until every account in the swept window has actually been	UNKNOWN / UNKNOWN	REJECTED	—
INFO	`H9-resolved-mode-mature-claim`	Once a market enters Resolved mode, no further accrual of claimable_pnl is possible against the residual; only existing	UNKNOWN / UNKNOWN	REJECTED	—

A — SEVERITY RUBRIC

TIER	DEFINITION
CRITICAL	Direct loss of user funds or full protocol takeover with no meaningful preconditions. Reachable from a permissionless instruction by any signer. Must be patched immediately.
HIGH	Significant loss of user funds or protocol invariant violation under realistic preconditions (specific market state, signer with limited but obtainable role). Patch should ship in next release.
MEDIUM	Hardening issue, partial loss possible, or invariant violation requiring privileged signer or improbable state. Worth fixing in normal cadence.
LOW	Minor issue with no plausible path to fund loss. Code-quality or defense-in-depth concern.
INFO	Informational. No security impact. Documentation or style suggestion.

B — METHODOLOGY

This cycle was produced by Jelleo's continuous, hypothesis-driven Solana audit loop. Every finding originates as a falsifiable invariant claim from a per-protocol hypothesis library, dispatched to multi-agent recon (Layer 1), promoted on contested verdicts via adversarial debate (Layer 1.5), and confirmed empirically via a `cargo test` proof-of-concept (Layer 2) before transitioning to `confirmed`. Confirmed findings auto-fire structural sibling derivation and cross-protocol propagation hooks, then move through a restricted lifecycle (`new → triaged → confirmed → disclosed → fixed → verified`). Every cycle is signed Ed25519 against the platform key — see the cover-page receipt.

Full spec: docs/methodology/ (eleven sections, §01–§10) · Live reference: jelleo.com/methodology.html · Inaugural disclosure: aeyakovenko/percolator-prog#39 (F7, 2026-04)