JELLEO | AUTONOMOUS SOLANA AUDIT

AUDIT REPORT · CYCLE 20260428-225814

Hunt cycle · default.

AUDITOR	Kirill Sakharuk · kirill@jelleo.com
CUSTOMER	default
WINDOW	cycle 20260428-225814
CYCLE	20260428-225814
ENGINE SHA	a946e5508f
WRAPPER SHA	d163960700
GENERATED	2026-05-08T22:32:57+00:00

0	0	0	0	0
CRITICAL	HIGH	MEDIUM	LOW	INFO

CONFIRMED · DISCLOSED · FIXED · VERIFIED

15 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

`20260428-225814` · started 2026-04-28T22:58:14+00:00 · engine `a946e5508f` · wrapper `d163960700`

01 — CYCLE SUMMARY

CRITICAL	HIGH	MEDIUM	CONFIRMED
2	8	5	0

TOTAL VERDICTS
15

Critical 2 · High 8 · Medium 5 · Low 0 · Info 0

02 — FINDINGS

SEVERITY	HYPOTHESIS	TITLE	VERDICT	STATUS	POC
CRITICAL	`CI1-deposit-then-withdraw-zero`	Deposit X immediately followed by withdraw X (with no intervening activity) leaves vault + account-state byte-identical	UNKNOWN / UNKNOWN	REJECTED	—
CRITICAL	`V1-vault-residual-conservation`	The post-haircut residual cash (vault - cash_locked_in_orderbook - claimable_pnl - insurance_counter) is conserved acros	UNKNOWN / HIGH	REJECTED	—
HIGH	`IX1-ix-data-validation`	Every instruction validates the length and shape of `instruction_data` before reading typed fields. No out-of-bounds rea	UNKNOWN / HIGH	REJECTED	—
HIGH	`L1-liquidation-discount-bounded`	Liquidation bonus paid to a liquidator cannot exceed the configured LIQUIDATION_INCENTIVE_PCT of seized collateral, even	UNKNOWN / MED	REJECTED	—
HIGH	`L5-liquidation-no-fee-enrichment`	Liquidation does not transfer collateral to the liquidator beyond the configured incentive percentage + protocol-defined	FALSE / MED	REJECTED	—
HIGH	`O3-position-authority-binding`	An account's `position_q` and `claimable_pnl` can only be mutated when the account's bound authority signs (or via permi	UNKNOWN / UNKNOWN	REJECTED	—

SEVERITY	HYPOTHESIS	TITLE	VERDICT	STATUS	POC
HIGH	`O4-im-respect-on-open`	Every instruction that opens or grows a position enforces initial-margin (IM) requirements: equity ≥ |position_q| × mark	FALSE / MED	REJECTED	—
HIGH	`P4-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 / MED	REJECTED	—
HIGH	`P9-pnl-arithmetic-bounds`	The lazy mark-to-market computation pnl_delta = abs_basis * (K_now - K_snap) / (a_basis * POS_SCALE) cannot overflow i12	UNKNOWN / UNKNOWN	REJECTED	—
HIGH	`V4-vault-cap-respect`	Vault balance is provably bounded by MAX_VAULT_TVL across every reachable state. No accounting helper can push vault pas	FALSE / HIGH	REJECTED	—
MEDIUM	`AC8-account-zeroing-on-close`	When an account is closed (via reclaim or full settlement), all its persistent fields are zeroed before the slot is mark	UNKNOWN / UNKNOWN	REJECTED	—
MEDIUM	`AR6-square-root-bounds`	Any sqrt-based computation (e.g., for vega-style adjustments) is bounded and never produces NaN-equivalents on integer a	FALSE / HIGH	REJECTED	—
MEDIUM	`AR7-saturating-arithmetic-correctness`	Where the codebase uses saturating arithmetic, the saturation point is the documented protocol cap, not a primitive type	UNKNOWN / HIGH	REJECTED	—
MEDIUM	`O10-orderbook-side-balance`	Total bid-side cash locked equals sum of (size × price) for all open bids; analogous for asks. Cannot be drained by help	UNKNOWN / UNKNOWN	REJECTED	—
MEDIUM	`O9-position-bedge-correct`	The "bedge" (basis-edge) accounting on partial closes correctly apportions realized PnL between the closed and remaining	FALSE / HIGH	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.

TIER	DEFINITION
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)