Phoenix5981 Security — Security Research Summary

Contact: willworachat@gmail.com | Report date: 2026-08-13 | Report type: Audit Coverage Summary

Target	Firelight — on-chain cover / insurance protocol (Solidity, Flare network)
Platform	Immunefi — Audit Competition (competition live 12–25 August 2026; $20,000 USDC pool)
Engagement dates	2026-08-13 (single-session)
Reports submitted	0
Outcome	**NO FINDINGS — CLEAN RESULT ACROSS 15 ANALYSED SURFACES**
Estimated coverage	Approximately 55% of the 4,253-line in-scope corpus by line count, but effectively complete coverage of the unprivileged attack surface: every externally reachable path not gated behind a trusted role was analysed to a definitive conclusion. The unread remainder is almost entirely privileged-path configuration code excluded by the programme's own trusted-party clause.

Scope Tested

Firelight is a period-based on-chain cover protocol deployed to the Flare network. Twelve Solidity files totalling 4,253 lines were in scope at competition-pinned commit `42f9ea5e` (branch `v1_audit_ready`): an upgradeable ERC-4626 vault with delayed, per-period withdrawal queues (1,133 lines); a cover-order allocator implementing order lifecycle and Merkle-committed settlement against an off-chain matching engine (1,012); an incident manager performing loss assessment and payout execution (908); a historical vault-accounting checkpoint library (403); a Flare FTSO v2 price adapter presenting a Chainlink `AggregatorV3` surface (251); a reward distributor, a receipt NFT, and three small conversion and validation libraries.

The protocol's defining mechanism is loss socialisation. Underwriters deposit into the vault; withdrawals burn shares immediately but enqueue assets into a per-period pool claimable only after that period closes. When an incident is assessed, payouts draw proportionally from both currently active assets and from withdrawal queues still considered exposed, so exiting underwriters cannot escape a loss they were underwriting at the time of capture.

Methodology Applied

- **Trust-graph mapping before code review.** The programme excludes findings requiring trusted parties to act maliciously within documented privileges, but explicitly re-admits them where a role can exceed its intended authority. The nine vault roles, four allocator roles, the off-chain matching engine, and the premium-conversion and first-loss-buffer counterparties were therefore enumerated first, so effort could be aimed at the unprivileged surface and at authority boundaries rather than at admin capability.
- **Algebraic verification of value-conservation arithmetic rather than inspection.** The incident payout routine splits a payment three ways across active assets and two withdrawal queues with a rounding-correction step. Rather than reading it for plausibility, the split was proved: that the active-assets component cannot exceed available active assets, that the vault's solvency expression cannot underflow, and that the correction step's reassignment is bounded by the destination queue.
- **Directional analysis of every rounding site.** Each conversion was assessed for which party it favours, including the chained conversions in incident payout pricing and the virtual share and asset offsets applied to per-period withdrawal pools, where the aggregate bias was bounded quantitatively rather than dismissed.
- **Window-disjointness proofs for state-machine races.** Where two operations could plausibly overlap in time — incident payout against withdrawal claiming, and permissionless expiry-cancellation against settlement — the guarding conditions were compared directly to establish disjointness rather than assuming the intended ordering holds.
- **Verification of candidate findings against source before write-up.** Two plausible issues and one strong lead were each carried to a conclusion that eliminated them, in every case by reading the surrounding source rather than by reasoning from the pattern alone.

Coverage

All externally reachable non-privileged behaviour was analysed. In the vault: deposit, mint, withdraw, redeem, withdrawal-request enqueueing, claim, and the incident payout routine, together with the ERC-4626 conversion layer and the solvency relationship between the vault's live asset balance and its pending-withdrawal liability. In the allocator: the single permissionless entry point, the Merkle-committed settlement path including per-market, per-protocol-concentration and aggregate capacity caps, and the ERC-721 mint callback reachable by a contract counterparty. In the oracle layer: the FTSO adapter's read surface

and the shared price-validation library covering positivity, round consistency, and staleness. In the accounting layer: the custom 256-bit checkpoint indirection that supplies the historical exposure cap bounding every incident payout.

Not analysed, and disclosed as such: cover-order creation, allocation commitment, and capacity configuration (all role-gated), the reward distributor, the receipt NFT, and the assessment half of the incident manager — approximately 2,000 lines, substantially all of it privileged-path code falling under the programme's trusted-party exclusion.

Outcome / Findings Summary

No vulnerabilities of any severity were identified. Notable negative results, each independently verified:

- **Incident payout arithmetic proved sound.** The three-way split conserves value exactly, the vault's solvency expression cannot be driven negative, and the rounding-correction step cannot over-assign to the destination queue. The last of these was the most plausible defect in the routine and was disproved algebraically rather than by testing.
- **Loss socialisation is correct and cannot be escaped.** Reducing a period's withdrawal assets without reducing its withdrawal shares dilutes that period's claimants pro rata, as intended. The payout window and the earliest claim window for every affected period are provably disjoint, so no queued underwriter can claim ahead of a payout that is meant to reach them. Withdrawals enqueued before an incident are correctly outside the exposure cap, since their assets have already left the vault's active accounting.
- **A donation-based share-inflation lead was identified and eliminated.** The vault computes total assets from a live token balance and does not raise the inherited virtual-share offset, which together form the standard precondition for first-depositor inflation. The attack nonetheless cannot extract value, because the deposit path explicitly rejects deposits that would round to zero shares — the victim's transaction reverts rather than exchanging assets for nothing. The residual is an unprofitable griefing variant.
- **The off-chain matching engine cannot under-charge premium.** Settlement verifies a double-hashed Merkle leaf against a per-period committed root, and the premium is recomputed on-chain from the allocated cover and the stored rate rather than being read from the proof, so a compromised matcher cannot decouple premium from cover. Capacity and concentration caps are evaluated against immutable committed state, which also contains the ERC-721 mint callback reachable mid-batch by a contract buyer.
- **The custom historical-accounting layer holds.** The checkpoint library stores 256-bit values in a side array indexed from an underlying 208-bit trace, using index zero as a combined sentinel and not-found marker. The index-to-value invariant was verified across insertion, same-key overwrite, and removal — removal being the natural place for the two structures to desynchronise and silently corrupt every historical lookup, including the exposure cap. The hinted lookup validates a caller-supplied hint before trusting it and otherwise falls back to a full search, so an adversarial hint cannot yield an incorrect value.
- **Two candidate issues were eliminated as documented behaviour.** The first-loss-buffer pull limits itself against the payer's balance but not their allowance, so an under-approved buffer reverts an entire incident payout — an inconsistent defensive posture, but one the source explicitly designates as intentional in two separate comments. Separately, a fee-on-transfer buffer token would understate the residual loss and short a claimant, but the code states the token is assumed to be a stablecoin, placing it outside the applicable scope gate for non-standard tokens.

$0. No reports submitted to the platform. The engagement was concluded on an expected-value judgement: the unprivileged surface was covered without result, the codebase showed consistent evidence of prior adversarial review — including comments pre-empting several of the specific attacks constructed here — and the remaining unread code is predominantly privileged-path logic excluded by the programme's trusted-party clause, assessed against a $20,000 pool distributed by rank across a competitive field.