Silent-Alarm Remediation Wave — MC #100785 follow-ups (#106201-106204)

Silent-Alarm Remediation Wave — MC #100785 follow-ups (2026-07-23)

Fixes for the silent/blind-spot alarm findings from the alarm audit (`~/system/evidence/100785/alarm-notification-audit-2026-07-22.md`, BookStack link/3254). All three are internal-daemon patches on ANVIL; each reuses existing cooldown infra so it **cannot** recreate the 2026-05-15 Slack-flood incident. All independently verified by John against the live working-tree files (the LaunchAgents execute the working tree, so the fixes are live without redeploy).

Shared constraint honoured by every fix: alerts are **cooldown-gated** (≥60-min window, `alert-gate.js` / per-daemon state-file convention), routed to `#alerts` (not `#exec`/`#ops`), with durable state files under `~/system/state/` (never `/tmp`).

#106201 — Health tooling now covers `com.alai.*`

Gap: `daemon-health.js` / `health-check.js` only enumerated `com.john.*`. The entire `com.alai.*` namespace (61 plists) was invisible — the biggest structural blind spot in the audit.

Fix: `daemon-health.js` — new `DAEMON_PREFIXES = ['com.john.', 'com.alai.', 'no.alai.']` used across `getLaunchAgentStatus/getPlistFiles/getParkedDaemons/isProcessActuallyRunning`. `health-check.js` — `SERVICES.daemons` watchlist widened 7→13 (adds the 6 audit-named degraded `com.alai.*` daemons).

Verified (John, live): both `node --check` OK; `daemon-health.js --json` coverage total 107→168, `com.alai.* 0→61`, degraded 1→3; `com.alai.email-ingest-monitor` + `com.alai.ollama-serve-v2` now surface as degraded. Evidence `~/system/evidence/106201/`.

Caveat: `health-check.js --quick` is HTTP-only by design and never runs the daemon check (true before and after) — intent verified via `daemon-health.js --quick`, which does show the widened set. Recommend a light Proveo regression on `ops-watchdog.js` + `mc-dashboard.js` (both consume `daemon-health.js`) with the larger 168-row set.

#106202 — `intake-classifier-sweep.sh` FAIL now alerts

Gap: failing every ~20 min (1216 accumulated silent FAILs), log-only, no notify path.

Fix: on FAIL, one cooldown-gated `slack.js send alerts` with fail-count + last-error snippet; `COOLDOWN_SECONDS=3600`; durable state `~/system/state/intake-classifier-sweep-lastalert.ts`; `RECOVERY` line + state clear on next success.

Verified (John, live): `bash -n` OK; cooldown/suppress/recovery paths present; state file durable. Evidence `~/system/evidence/106202/` (fire-then-suppress simulation).

Caveats (must-know):

- `set -euo pipefail` → `set -uo pipefail` (dropped `-e`) so the alert fallback runs; all FAIL branches still `exit 1`, so LaunchAgent-visible exit behaviour is unchanged.
- The classifier's **root cause is NOT fixed** (out of scope): `intake-classifier-deterministic.js:184` `writeFileSync('/tmp/evidence-104025/…')` with no `mkdir -p` → `ENOENT` every run. So **expect a real `#alerts` message shortly after the next cycle** — that is the fix working (silent alarm now speaks), not a new problem. Follow-up: add `mkdir -p` (audit rec #2).

#106203 — `seo-intake-watcher.js` self-failure alert

Gap: crash-looped ~21h on Kudu VFS 403 (2026-07-22), self-recovered, nobody alerted. Emailed on new submissions but had no alert path for its own repeated failure.

Fix: persistent consecutive-failure counter (`~/system/state/seo-intake-watcher-failstate.json`); alert once a streak crosses `FAIL_STREAK_THRESHOLD=3` (~30 min sustained), then `ALERT_COOLDOWN_MS=60min` between repeats; one-line recovery on next success **only if** the streak had

alerted; sub-threshold blips reset silently (no flood on transient blips). Channel `alerts`. Submission-email + AAD/Kudu auth logic untouched.

Verified (John, live): `node --check` OK; `recordPollFailure/Success` present (L107-154); cooldown gating real. Evidence `~/system/evidence/106203/` (4-scenario simulation ALL PASSED).

Cross-cutting notes

- All three committed on branch `deploy/memory-p1-shadow-20260723` (pre-existing stray branch; a `~/system` auto-backup daemon swept edits into backup commits mid-session — per-file diffs verified to contain exactly the intended change). Working-tree files are live; a clean main-merge is separate hygiene.
- Remaining audit follow-up: **#106204** (M) — `litestream-staging-prune.sh` STALL + `reality-anchor-watchdog.sh` → Slack, plus a PAT/token-expiry monitor. Plus audit rec #7 (generic consumer for `daemon-fail-counters/`) still open/unassigned — it would auto-catch this whole bug class.

Revision #1
Created 2026-07-23 13:06:03 UTC by John
Updated 2026-07-23 13:06:04 UTC by John