

mini-verifier canary — dnevni known-answer monitoring (MC #105136)

MC #105136 — mini-verifier canary in auto-verify-regression — Canary Report

Scope

Extended `~/system/tools/auto-verify-regression.js` with a new "mini-verifier canary" section (3 known-answer cases run daily through `~/system/tools/mini-verifier.js`, the pi-orch \$0/call local-LLM closure gate). Approved by CEO 2026-07-09 as a 30-day monitoring pilot (#105107/#105113).

Files touched

- Edited: `/Users/makinja/system/tools/auto-verify-regression.js` (only file edited; no new LaunchAgent, no edits to `mini-verifier.js`, no edits to the existing 5 claim-verifier TESTS)
- New (evidence only, not production): `/Users/makinja/system/evidence/105136/*`
- Production monitoring feed: `/Users/makinja/system/state/mini-verifier-canary.jsonl`

AC Checklist

#	Acceptance criterion	Result
---	----------------------	--------

1	New "mini-verifier canary" section added after the existing 5 slučajeva	PASS — <code>CANARY_CASES</code> array + <code>runCanaryCase</code> / <code>runCanarySection</code> defined after <code>TESTS</code> / <code>runTest</code> , invoked in <code>main()</code> after the claim-verifier loop
2	C1: true claim about existing file (<code>mc.js</code>) with that file as evidence → expect CONFIRMED	PASS — verdict CONFIRMED, matched
3	C2: claim about nonexistent file + same nonexistent path as evidence → expect HALLUCINATION	PASS — verdict HALLUCINATION, matched
4	C3: claim about nonexistent file + unrelated <i>existing</i> evidence (BUILD-BLUEPRINT.md, #105113 repro) → expect HALLUCINATION	PASS — verdict HALLUCINATION, matched
5	Mismatch or SKIP on C1-C3 → same Slack alert mechanism reused	PASS (verified in V2) — reuses existing <code>sendSlackAlert()</code> → <code>slack.js</code> <code>send ai-ops</code> ; SKIP does NOT alert (treated as WARN per spec), only true verdict mismatch alerts
6	Every run appends JSON line to <code>~/system/state/mini-verifier-canary.jsonl</code> (ts, case, expected, actual, reasons, latency_ms, model_id)	PASS — all 6 fields present, verified in V1 output
7	Canary section does not break the 5 existing claim-verifier cases or change their exit-code semantics	PASS — 5/5 claim-verifier PASS unaffected in both V1 and V2 runs; <code>allPassed</code> (claim-verifier) logic untouched, only OR'd with new <code>canaryHasFail</code> for overall exit code
8	No new LaunchAgent	PASS — only pre-existing <code>com.john.auto-verify-regression.plist</code> present, unchanged, same script path (V3)
9	No edits to <code>mini-verifier.js</code>	PASS — file untouched (only read for interface discovery)
10	New files TS-only	N/A — no new <i>production</i> files were created; the only new "files" are evidence artifacts (backup .js copy, .txt outputs), not new production source files. The one edit is to the existing <code>.js</code> production file per explicit permission ("Editovanje postojećeg .js dozvoljeno")
11	No secrets in output	PASS — grep scan of all evidence files for api-key/secret/token/password/bearer patterns returned nothing

V1 — Full manual run (production script, unmodified expectations)

Command: `node ~/system/tools/auto-verify-regression.js --verbose` Exit code: **0** Result: **5/5 claim-verifier PASS + 3/3 canary OK** Raw stdout: `V1-full-run-stdout.txt` (stderr empty: `V1-full-run-stderr.txt`) jsonl after run (3 new lines, all status=OK): `V1-canary-jsonl-after-run.txt`

```
AUTO-VERIFY REGRESSION: 5/5 PASS
  v Tunnel count mismatch -> correctly FAIL
  v Domain existence false negative -> correctly FAIL
  v Email account count -> correctly FAIL
  v Live domain status -> correctly PASS
  v Domain exists -> correctly PASS
MINI-VERIFIER CANARY: 3/3 OK
  v mini-verifier canary: true claim, existing evidence -> correctly CONFIRMED
  v mini-verifier canary: hallucinated claim, nonexistent evidence -> correctly HALLUCINATION
  v mini-verifier canary: hallucinated claim, unrelated existing evidence (#105113 repro) ->
correctly HALLUCINATION
```

V2 — Alert path validation (TEMPORARY test copy only, production untouched)

A temporary copy `auto-verify-regression.TEST-COPY.js` was made in the evidence dir (never in `~/system/tools/`). In that copy ONLY: C1's `expected` was flipped from `CONFIRMED` to `HALLUCINATION` (intentionally wrong, to force a mismatch), and `sendSlackAlert()` was given a `CANARY_TEST_DRY_RUN=1` env-gated dry-run branch that logs the exact command instead of executing it — so the real `#ai-ops` channel was NOT spammed with a synthetic test alert.

Command: `CANARY_TEST_DRY_RUN=1 node .../auto-verify-regression.TEST-COPY.js --verbose` Exit code: **1** (as expected — proves overall exit code goes non-zero on canary FAIL) Result: 5/5 claim-verifier PASS (unaffected) + **2/3 canary OK, 1 FAIL (C1, forced)** Raw stdout: `V2-alert-path-stdout.txt`

Key excerpt proving the alert branch fired:

```
WARN:   FAIL: mini-verifier canary: true claim, existing evidence — got CONFIRMED, expected
HALLUCINATION
```

```
MINI-VERIFIER CANARY: 2/3 OK
```

```
x mini-verifier canary: true claim, existing evidence -> UNEXPECTED: got CONFIRMED, expected HALLUCINATION
```

```
INFO: [DRY-RUN] would exec: node .../system/tools/slack.js send ai-ops "[ALERT] mini-verifier canary FAILED: mismatch on known-answer case(s)
```

```
Failures:
```

```
- mini-verifier canary: true claim, existing evidence: got CONFIRMED, expected HALLUCINATION  
mini-verifier.js may be broken. Check: /Users/makinja/system/state/mini-verifier-canary.jsonl"
```

This confirms: (a) the exact same `sendSlackAlert()` → `slack.js send ai-ops` call path used by the 5 existing claim-verifier cases is reused for canary failures, (b) the overall script exit code becomes 1 when a canary case fails even if all 5 claim-verifier cases still pass.

Cleanup honesty note: because `CANARY_JSONL` path was not overridden in the test copy, this V2 run also appended 3 lines (including one synthetic FAIL for C1) to the real production `~/system/state/mini-verifier-canary.jsonl`. This was caught and corrected: the full polluted file was saved to `V2-canary-jsonl-BEFORE-cleanup.txt` for the record, then the production jsonl was restored to contain only the 3 genuine V1 lines (verified via diff against `V1-canary-jsonl-after-run.txt` before truncating). The 30-day monitoring feed is clean of test artifacts as of this report.

V3 — LaunchAgent unchanged

`~/Library/LaunchAgents/com.john.auto-verify-regression.plist` inspected: still the only relevant plist, `ProgramArguments` still points at `/Users/makinja/system/tools/auto-verify-regression.js` (same path, untouched schedule `StartInterval=900`, `RunAtLoad=true`). No new LaunchAgent files present. Saved to `V3-launchagent-plist.txt`.

Diff summary

`diff-pre-vs-post.txt` — 251 diff lines, all additive:

- Updated file header comment to mention the canary section.
- Added `MINI_VERIFIER` and `CANARY_JSONL` path constants.
- Added `CANARY_CASES` array (C1/C2/C3), `runCanaryCase()`, `appendCanaryJsonl()`, `runCanarySection()`.
- In `main()`: after the existing claim-verifier report/log block, added a call to `runCanarySection()`, a canary report print block, and extended the alert condition to `!allPassed || canaryHasFail` (was `!allPassed` only) with an appended canary-specific alert message segment when applicable.
- Zero lines removed/modified inside the original `TESTS` array, `runTest()`, or `sendSlackAlert()` in the production file (the dry-run branch only exists in the disposable

What was NOT done (explicitly out of scope, confirmed absent)

- No new LaunchAgent plist created.
- No edits to `mini-verifier.js`.
- No edits to the 5 existing claim-verifier test cases or their pass/fail logic.
- No secrets printed to stdout/stderr/evidence files (checked).

Evidence paths (all absolute)

- `/Users/makinja/system/evidence/105136/auto-verify-regression.js.pre-105136` (pre-edit backup)
- `/Users/makinja/system/evidence/105136/diff-pre-vs-post.txt`
- `/Users/makinja/system/evidence/105136/V1-full-run-stdout.txt`, `V1-full-run-stderr.txt`, `V1-exit-code.txt`
- `/Users/makinja/system/evidence/105136/V1-canary-jsonl-after-run.txt`
- `/Users/makinja/system/evidence/105136/auto-verify-regression.TEST-COPY.js` (disposable, alert-path test only)
- `/Users/makinja/system/evidence/105136/V2-alert-path-stdout.txt`, `V2-alert-path-stderr.txt`, `V2-exit-code.txt`
- `/Users/makinja/system/evidence/105136/V2-canary-jsonl-BEFORE-cleanup.txt` (honest record of the test-copy jsonl pollution, since corrected in production)
- `/Users/makinja/system/evidence/105136/V3-launchagent-plist.txt`
- `/Users/makinja/system/evidence/105136/canary-report.md` (this file)
- Production: `/Users/makinja/system/tools/auto-verify-regression.js` (edited)
- Production: `/Users/makinja/system/state/mini-verifier-canary.jsonl` (clean, 3 genuine lines from V1)

Evidence: ~/system/evidence/105136/ | Monitoring feed: ~/system/state/mini-verifier-canary.jsonl | Kadenca: svakih 15 min (StartInterval=900, com.john.auto-verify-regression) | Lanac: #105018 → #105104 → #105107 → #105113 → #105136 | P2P mesh: mesh-thr-439d874c / mesh-msg-2924726d | 2026-07-09

Revision #1

Created 2026-07-09 19:49:21 UTC by John

Updated 2026-07-09 19:49:21 UTC by John