OCD-Delta Webhook Runbook — Anti-Hallucination V2

OCD-Delta Webhook Runbook — Anti-Hallucination V2

Component: OCD-Delta (Orchestrator Claim Detector — Delta)
Source spec: Anti-Hallucination V2 §4 (Secondary Hardening)
MC: #99732
Published: 2026-05-22

Overview

The OCD-Delta webhook fires on Task tool PostToolUse. It detects verdict claims in agent text output and blocks propagation if the verdict does not satisfy the V2 contract. This closes the gap where Proveo text claims PASS but the orchestrator accepts the claim before any gate fires (MC #99595 failure mode).

Trigger

- **Hook type:** PostToolUse
- **Scoped to:** Task tool responses
- **Fires when:** A Task tool call returns text containing verdict keywords (PASS, FAIL, GO-LIVE-READY, PARTIAL, BLOCKED, REFUSED)

Blocking Conditions

OCD-Delta blocks (exits non-zero) when ANY of:

- `evidence_files` absent or empty array
- `machine_checks_executed` < `machine_check_count`

- `expires_at` absent
- `expires_at` is in the past (TTL expired)
- Verdict is GO-LIVE-READY and `john_reproducer_output` absent
- Verdict is GO-LIVE-READY and quorum count < 2

Workaround for PostToolUse Limitation

Claude Code does not expose raw Task response text to bash hooks directly. Protocol:

1. Agent writes verdict JSON to `/tmp/ocd-delta-manifest-<mc_id>.json` before returning its response
2. OCD-Delta reads this manifest file
3. Hook validates and exits 0 (allow) or 1 (block)
4. If manifest absent: hook logs warning and allows (backward-compatible)

Verdict TTL Check

```
EXPIRES_AT=$(jq -r .expires_at /tmp/ocd-delta-manifest-<mc_id>.json)
NOW=$(date -u +"%Y-%m-%dT%H:%M:%SZ")
if [[ "$NOW" > "$EXPIRES_AT" ]]; then
  echo "ERROR: Verdict expired at $EXPIRES_AT. NULL verdict — rerun required."
  exit 1
fi
```

Machine Check Count Validation

```
COUNT=$(jq .machine_check_count /tmp/ocd-delta-manifest-<mc_id>.json)
EXECUTED=$(jq .machine_checks_executed /tmp/ocd-delta-manifest-<mc_id>.json)
if [[ "$EXECUTED" -lt "$COUNT" ]]; then
  echo "ERROR: machine_checks_executed ($EXECUTED) < machine_check_count ($COUNT). Verdict
invalid."
  exit 1
fi
```

GO-LIVE-READY Quorum Check

```
VERDICT=$(jq -r .verdict /tmp/ocd-delta-manifest-<mc_id>.json)
if [[ "$VERDICT" == "GO-LIVE-READY" ]]; then
  MATCHES=$(jq -r .john_reproducer_output.matches_verdict /tmp/ocd-delta-manifest-
<mc_id>.json)
  if [[ "$MATCHES" != "true" ]]; then
    echo "ERROR: GO-LIVE BLOCKED — john_reproducer_output.matches_verdict is not true. ZAKON
#29.2 violation."
    exit 1
  fi
fi
```

Installation

1. Script: `~/.claude/hooks/ocd-delta-validator.sh`
2. Register in Claude Code settings as PostToolUse hook scoped to Task tool
3. Make executable: `chmod +x ~/.claude/hooks/ocd-delta-validator.sh`
4. Test: create `/tmp/ocd-delta-test.json` with missing `evidence_files`, run hook, expect exit 1

Monthly Hallucination Drill

LaunchAgent: `com.alai.hallucination-drill`
Plist: `~/Library/LaunchAgents/com.alai.hallucination-drill.plist`
Schedule: monthly

Drill sequence: generate synthetic verdict with PASS claim but false intent_proof → run through OCD-Delta → expect HALLUCINATION_DETECTED (exit 1). If hook exits 0: auto-create P0 MC, block all H/BLOCKER task closes until patched.

Escalation

When blocked: print error to stderr with MC ID and reason. If verdict=REFUSED: auto-post to Slack #john-alerts within 15 minutes. Suspend all dependent task completions until CEO arbitrates.

Source: Anti-Hallucination V2 §4 | MC #99732 | Cross-ref: BookStack page 2995 (full spec)

Revision #1
Created 2026-05-22 08:29:34 UTC by John
Updated 2026-05-22 08:29:34 UTC by John