HITL Review Policy

Defines when human reviewers must be in the loop, what they can and cannot override, and the exact override event payload (contract C2) that downstream provenance modules consume. This policy is the operating contract between the agent runtime and the human reviewer pool.

1. Mandatory HITL gates

A run MUST route through an HITL gate before execution when any of the following is true:

- The planned action crosses the irreversible_impact_boundary declared on the Operating Context Canvas.
- confidence_band is low for any action class, or medium for any action with reversibility = irreversible or partially-reversible.
- Any of: unsafe-action-attempted, pii-leak-risk, prompt-injection-detected, false-success-report, hallucinated-citation fired in the current run.
- The agent reached Escalated per the Fallback Blueprint.
- The agent has accumulated two or more failure_id events of severity high or critical in a rolling 7-day window for the same agent.

A run MAY route post-hoc through HITL review (sampling) when:

- confidence_band is medium and action is reversible (random 10% sample, plus all overrides chains).
- The agent is in canary or percentage-rollout stage (per s1-monitoring-rollout-postmortem.md).

2. Reviewer authority

Reviewers may:

- Approve as proposed.
- Permit-with-modification (edit the staged output, retain plan).
- Refuse-and-instruct with a one-shot corrective instruction valid for the current run only.
- Escalate to the next tier per the threshold spec.
- Open a hardening task in the backlog.

Reviewers may NOT:

- Edit the irreversible_impact_boundary list on the canvas. Boundary changes require an S1 owner version bump.
- Change a failure-mode severity value or rename a failure_id.
- Lift a pii-leak-risk block alone. That requires a compliance role plus reroute to an authorized sink.

- Bypass the override event logging. Any reviewer decision that does not emit a complete event is rejected by the runtime.

3. Override event payload schema (contract C2)

S3 provenance consumes this exact shape. Field names and enums are locked. Additive fields require coordination with S3.

```
{
  "event_id": "uuid-v7",
  "timestamp": "ISO-8601 UTC with millisecond precision",
  "agent_id": "kebab-case agent id from Operating Context Canvas",
  "decision_origin": "agent | human-override | fallback | escalation",
  "evidence_pointer": "uri to evidence bundle (provenance store)",
  "rationale": "free-text reviewer rationale; required when decision_origin=human-override
or escalation",

  "run_id": "uuid-v7 of the current run",
  "step_id": "id of the agent step the override applies to",
  "reviewer_role": "role name; not personal identity",
  "reviewer_id_hash": "salted hash of reviewer principal id",
  "action_type": "approve | permit-with-modification | refuse-and-instruct | escalate |
hardening-task-opened",
  "rationale_code": "controlled-vocabulary code, see appendix",
  "failure_id_refs": ["zero or more failure_ids from the register that this override
addresses"],
  "before_state_hash": "hash of staged action and inputs",
  "after_state_hash": "hash of approved action and inputs after modification",
  "sla_target_ms": 0,
  "sla_actual_ms": 0,
  "policy_version": "semver of this policy at decision time",
  "canvas_version": "semver of the s1-operating-context-canvas at decision time"
}
```

The first six fields (event_id, timestamp, agent_id, decision_origin, evidence_pointer, rationale) are the contract C2 minimum. Runtime rejects any event missing these. The remaining fields are required by this policy but are additive from S3's perspective.

decision_origin enum is exhaustive:

- agent - autonomous action, logged for completeness when HITL sampling captures it.
- human-override - reviewer changed or modified the agent's plan.
- fallback - deterministic fallback path used (no reviewer involved).
- escalation - decision was made at a higher tier (Tier 2 or Tier 3 reviewer).

4. Rationale code vocabulary (appendix)

Controlled, append-only. Bumping requires S1 owner sign-off.

- RC-EVIDENCE-INSUFFICIENT - agent lacked grounding to justify the action.
- RC-BOUNDARY-CROSS - planned action touched the irreversible-impact boundary.

- RC-LOW-CONFIDENCE - confidence_band was low or unknown.
- RC-DRIFT-SUSPECTED - output quality looked anomalous vs. baseline.
- RC-INJECTION-RISK - untrusted content present in inputs.
- RC-PII-EXPOSURE - data class did not match destination authorization.
- RC-POLICY-CHANGE - upstream policy or statute changed; agent not yet updated.
- RC-OPERATOR-PREFERENCE - reviewer judgment; not a defect.
- RC-MODEL-ERROR - clear model mistake (hallucination, math error).
- RC-CONNECTOR-FAILURE - tool or connector returned bad data.

Use RC-OPERATOR-PREFERENCE sparingly. High volume of this code is a sign the prompt or canvas, not the reviewer, needs updating.

5. From recurring overrides to hardening tasks

The policy treats reviewers as signal generators, not patch monkeys. When a pattern emerges, it must be hardened.

Triggers for a hardening task:

- The same failure_id plus same rationale_code fires 3 times in 14 days on the same agent.
- Reviewer override rate exceeds the canvas-declared ceiling (default 8% rolling 7-day) for any agent.
- Median sla_actual_ms exceeds sla_target_ms for any tier over a rolling 14-day window.
- A permit-with-modification chain shows a recurring edit pattern detectable by simple diff clustering.

Hardening task types, in order of preference:

1. Deterministic fallback - add a rules-based path that resolves the case without the model.
2. Threshold tightening - raise the band cutoff or add a structured guardrail.
3. Prompt or tool change - last resort, requires an S2 eval delta to pass.
4. Canvas edit - if the underlying scope or constraint was wrong.

Hardening tasks are tracked in the agent's hardening backlog and surface in s1-monitoring-rollout-postmortem.md as part of the weekly reliability review. Each hardening task references the override events that motivated it, closing the loop between human review and automated improvement.

6. Reviewer ergonomics

A reviewer must see: the staged action, the top three evidence pointers, the failure_id refs, the relevant section of the canvas (objective + boundary list), and a single approve / modify / refuse / escalate control. Anything more is a sign the gate is misplaced or the agent is doing too much in one step.