TruVerifAI

PANEL REVIEW · EXECUTIVE OVERVIEW

Four models argue over your agents' riskiest changes, before they ship

For an evaluator or sponsor deciding whether Panel Review is worth their team's time.

The problem

AI agents now write most new code: Google reports ~75% of its new code is AI-generated; Anthropic, that Claude authors 80%+ of its merged production code. Adoption is near-universal (84% of developers, Stack Overflow 2025). But no single reviewer catches everything at that pace: a person cannot keep up with the volume of AI-generated code, and any one model has blind spots. That's where the risky changes slip through: a removed auth check, a destructive migration, a subtle logic bug that reads fine.

What Panel Review is

Two halves that work together:

1. **A multi-model review panel.** Four frontier models (across providers) review a change **independently**, then conflict-target each other's positions, and return a single **decision-grade** verdict with severity-tagged findings, not four opinions to reconcile yourself.
2. **Local review gates.** Hook code next to your coding agent **intercepts risky writes and commits before they land** and routes them to the panel. High-stakes changes (auth, secrets, money, migrations, a removed guard, or **your own** domain-critical code, via custom floors) don't ship unreviewed. The gate blocks; the panel decides; you proceed with evidence.

It runs across Claude Code, Codex, Cursor, Copilot/VS Code, Gemini, Antigravity, and as a git pre-commit hook: the same review on every surface your team uses.

Why it's different

- **Cross-vendor blind-spot review.** Four models from four different families catch what any single reviewer, human or model, misses.
- **It blocks before the merge, not after.** The gate is the point: risky changes are stopped at the write/commit, not flagged in a PR nobody reads.
- **You mark what matters.** Custom floors let you declare your own critical code ("our pricing engine", "our tax rules") so it always gets a real review.

- **Your data, your call.** Turn off content storage, bring your own model keys, or run the whole review on **your own models in your own cloud**, so the review never has to leave your tenancy.

The proof: we build it with it

From our own production telemetry building TruVerifAI with the gates armed (ten weeks, 479 review calls):

66%	76%	<1%	151
of reviews changed the agent's decision	decision-change rate on security reviews	of audits passed completely clean	escalations to a human: the moments that most needed a person

Try it without touching a real machine

A disposable sandbox (a GitHub Codespace in your own org, or your own AWS box) arrives with the gates pre-armed. Point it at your own non-critical code, watch it block a risky change in the first minute, and see the panel catch things over a few days. Delete the machine and nothing remains. Compute is a few dollars at most (often free); the evaluation credits are on us.

> **Open source & verifiable:** the entire client is MIT-licensed, zero runtime dependencies, published with provenance, and readable at github.com/TruVerifAI/init before you trust it with anything. The hosted panel is the only part that isn't open.

Companion documents: the **Security & Data-Flow** brochure (for your security team) and the **Sandbox Setup Guide** (for your engineers). Source: truverif.ai/panel-review.