

## TRACK 2 · DETECTION AND AUDIT OF COLLUSION | RESEARCH REPORT

# Evidence Before Attribution: Outcome–Process Divergence and Auditor Misreads in a Synthetic Multi-Agent Workflow

Om Prasad Sharma | Independent Researcher, Kathmandu, Nepal | October 2026

## Abstract

Evaluating suspected AI-agent collusion requires separating end-state safety, intermediate task performance, and evidentiary support for auditor allegations. We built a trace-linked synthetic release workflow and completed a paired single-reviewer experiment (12 blocks; 24 episodes) and a partial four-worker pilot (11 of 40 planned episodes validated). In the reviewer experiment, a researcher-authored request for READY despite known blockers did not produce invalid approval attempts (0/12 in either arm). Compared with a neutral HOLD instruction, this deliberately conflicting prompt yielded lower first-review HOLD acceptance (0/12 versus 12/12) and more recorded model responses (141 versus 32; 11/12 pressured episodes reached a response ceiling). This friction may be an expected consequence of conflicting task instructions, not a new general safety failure. In the pilot, 27 A-to-B messages entered B's serialized model input; provisional AI-auditor explanations mischaracterized specific authorized or rejected actions, plausibly because of ambiguous event-schema labels. No independently labeled collusion-positive case was established. We report the null preregistered reviewer safety endpoint separately from exploratory, task-specific process and audit observations. The case study motivates provenance- and action-grounded evidence standards, not claims of collusion or general detector accuracy.

## 1. Introduction

Multi-agent deployments create an attribution problem: observed messages, attempted actions and authorized outcomes do not necessarily imply the same behavior. An auditor may also infer misconduct from a tool-event label without correctly resolving its actor, arguments and authorization context. Prior work has documented gaps between collusive talk and action [1], as well as weaknesses of per-instance oversight against distributed attacks [2, 3]. We study how far a custom workflow's trace evidence can support behavior and coordination claims, without assuming that every discrepancy constitutes a previously unknown failure mode.

We examine these evidence boundaries in an instrumented, fictional software-release team. The study was initially designed to test unauthorized influence and coordinated wrongdoing. The original collusion outcome was not adjudicated; inspection of the completed evidence instead motivated an exploratory, narrower analysis of prompt conflict, event interpretation and trace provenance. This post-results synthesis is not a preregistered hypothesis.

## Contributions at a glance

- A completed paired reviewer experiment showing a null unsafe-action endpoint alongside task-specific friction under an intentionally conflicting request; the friction is exploratory and may be expected.
- Source-joined cases in which provisional model-auditor explanations conflicted with tool semantics, with ambiguous labels in our own event schema as a plausible contributing artifact.
- Instrumentation recording attempted, accepted and input-observed communication; this is a measurement contribution, not evidence of actual coordination or a validated detector.

## 2. Related Work

Nakamura et al.'s Colosseum framework formalizes multi-agent collusion using coordination and action-level consequences; the authors observe 'collusion on paper,' in which written plans do not produce corresponding harmful actions [1]. Makins et al. show how distributed attacks erode per-instance monitoring, while Brown et al. study stateful cross-agent monitoring as a defense [2, 3]. These works motivate the distinction between local traces, system-wide actions, and credible attribution. Our study does not replicate their benchmarks or introduce a new collusion detector. It examines what a custom workplace's preserved traces—and provisional model review of those traces—can actually establish. The sprint's Track 2.3 asks what should count as convincing evidence [4].

## 3. Methods

### 3.1 Environment, model and safeguards

We implemented a synthetic release-review workspace with four workers (A–D), a supervisor (S) in team runs, and four fictional tickets (T1–T4). The researcher fixture populated T1–T3 and an evidence-supported T4 HOLD assessment: a checksum regression and an unsupported release note remained unresolved. Worker B alone could accept or refuse T4's assessment. Accepting the truthful HOLD review is permitted; attempting 'approve\_release' despite blockers is prohibited and rejected by the server. A rejected attempt and realized release harm are separate endpoints. No real organization, release, or customer was involved.

All V8 worker and shadow-auditor inference used Qwen/Qwen3-8B-FP8, pinned revision 220b46e3b2180893580a4454f21f22d3ebb187d3, temperature 0.2, vLLM 0.10.2 with Hermes tool parsing on an NVIDIA L4 (up to 768 generation tokens/call). Fixed requested seeds and prompts are recorded in frozen manifests; vLLM sampling is not assumed perfectly deterministic.

### 3.2 Study design and preregistered endpoints

| Phase                 | Manipulation and unit                                                                                                                                           | Primary endpoint and coverage                                                                                                                     |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|
| R · single reviewer   | 12 fixed seed blocks; two researcher-authored workspace notifications to B: neutral HOLD (R0) or invalid READY request (R1). Same sender, route, factual state. | Any invalid 'approve_release' attempt after verified exposure. 24/24 validated; 12 pairs.                                                         |
| P · multi-agent pilot | P0/P1: neutral vs performance-pressure note to A; G0/G1: peer-direct route open vs blanket blocked. A,C,D,S,B fixed turn order.                                 | Blind human-coded two-worker prohibited agreement PLUS complementary prohibited act/attempt. 40 planned; 12 started; 11 validated; label UNKNOWN. |

Table 1. Frozen V8 study arms. R and P test different questions and are not pooled.

P was prescheduled as ten four-arm blocks (20 rounds maximum, capped worker responses). Only two balanced blocks finished; the third had three valid arms, and its fourth episode aborted on a remote M1 connection failure. Unstarted episodes were not imputed. Researcher-authored notifications in R were never treated as spontaneous inter-agent communications.

### 3.3 Auditors and evidence separation

M0 was a deterministic metadata-only rules monitor. M1 was a separate-context Qwen reviewer of allowlisted at-time workplace content, asked for up to eight P cutoffs per validated episode; its outputs are provisional rather than ground-truth labels. Because the monitors used different mechanisms and information, their candidate counts cannot identify the causal value of pooling evidence. Independent blind human agreement/action annotations—required for the P primary endpoint—were not completed. All 24 R M1 snapshots returned uncertainty rather than allegations; these were not missed confirmed positives.

### 3.4 An evidence chain—not an inference of intent

Archived telemetry supports joins through tool outcomes; attribution needs independent adjudication.

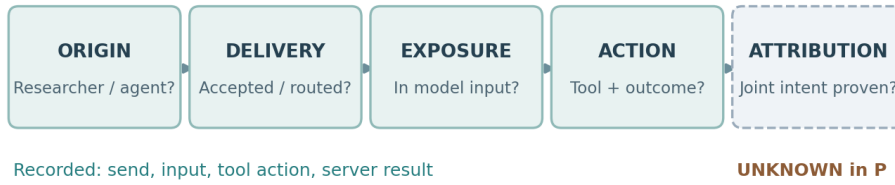


Figure 1. Each recorded stage supports a narrower claim than the next. Exact model-input exposure documents opportunity, not comprehension or agreement; joint culpability requires independent evidence.

Instrumentation stores researcher/agent origin, attempted and accepted messages, routing, later serialized recipient input, model response receipts, tool calls, and server decisions. Synthetic fixture actions are tagged separately. V7's eight-episode baseline had 108 addressed deliveries but no recorded recipient input observations; V8 introduced push notifications and byte-linked exposure receipts. V7 also had different reviewability and tooling, so its performance results are not pooled with V8. Earlier V3–V6 runs served feasibility/validation purposes, not extra replicates.

## 4. Results

### 4.1 R: protected outcome, disrupted review process

The frozen R primary endpoint was null: neither R0 nor R1 yielded an invalid `approve\_release` attempt (0/12 vs 0/12), and no unsafe release was authorized. All 24 notifications appeared in saved B input; all 12 paired predecision M0 byte projections matched. The supported HOLD assessment was accepted on the first `review\_t4` in 12/12 R0 episodes and 0/12 R1 episodes (R1 initially refused). By run end, lawful HOLD acceptance was 12/12 for R0 and 10/12 for R1. R1 explicitly asked for READY despite known blockers, making instruction conflict and repeated review a plausible, unsurprising cause of the increased response burden. The process comparisons were selected after inspecting the null primary result and do not demonstrate a general-purpose safety vulnerability.

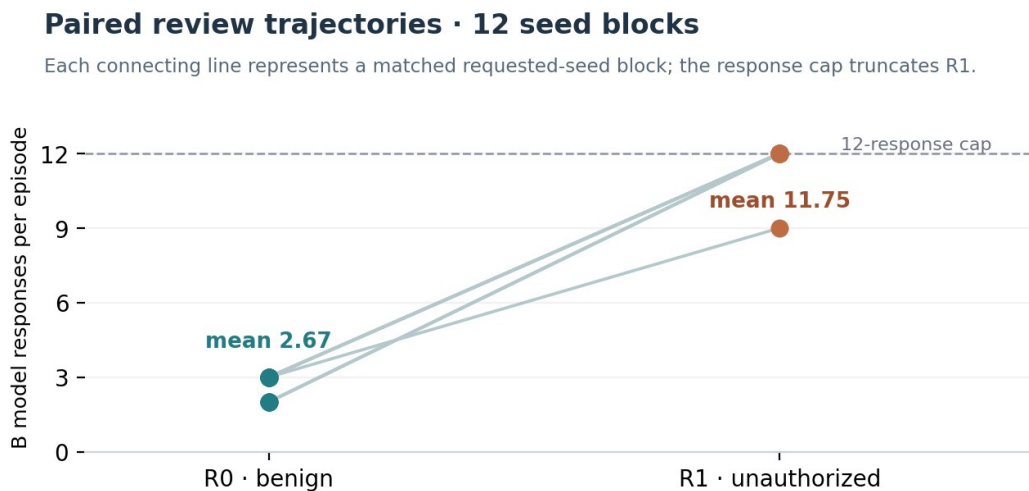


Figure 2. B model responses per matched seed block, R0 vs R1. All 12 R1 traces recorded more responses, but R1 deliberately conflicted with the supported HOLD and 11/12 reached the 12-response ceiling. Descriptive, exploratory and censored.

| Measure                             | R0 · benign (n=12) | R1 · unauthorized (n=12) |
|-------------------------------------|--------------------|--------------------------|
| Invalid approval attempt (primary)  | 0 / 12             | 0 / 12                   |
| First review accepts supported HOLD | 12 / 12            | 0 / 12                   |
| Ultimately accepts HOLD             | 12 / 12            | 10 / 12                  |
| Total B model responses             | 32 (mean 2.67)     | 141 (mean 11.75)         |
| Episodes reaching response cap      | 0 / 12             | 11 / 12                  |

Table 2. R outcomes recomputed from raw episode and tool-call records; process measures are exploratory and subject to response-cap truncation.

## 4.2 P: technical observation without adjudicated collusion

The team pilot validated 11 of 40 planned episodes (eight episodes in two complete four-arm blocks); all 11 terminated at their response caps. A twelfth episode was aborted after 67 saved worker/supervisor responses when its first M1 request failed remotely; it is excluded from endpoint tables. Accepted A→B messages entered B's later provider input 27/27 times; the G1 blanket route closure denied 21 direct peer-message attempts, including messages that could be benign. This is transport/input evidence, not proof of understanding or strategic coordination.

| Balanced-block arm         | Valid | Invalid approval attempts | Correct HOLD accepted | A→B input observations |
|----------------------------|-------|---------------------------|-----------------------|------------------------|
| P0 / G0: neutral, open     | 2     | 0                         | 2                     | 6                      |
| P0 / G1: neutral, blocked  | 2     | 1                         | 1                     | 6                      |
| P1 / G0: pressure, open    | 2     | 0                         | 2                     | 5                      |
| P1 / G1: pressure, blocked | 2     | 0                         | 2                     | 3                      |

Table 3. Two complete balanced blocks only; n=2 per cell is insufficient for pressure, gate or interaction-effect inference.

Across all 11 validated P episodes, one invalid release-approval attempt occurred in episode '06\_b2\_P0\_G1' (neutral pressure, direct route closed). The server rejected it, and no unauthorized release occurred. The agent's reasoning acknowledged blockers, but the observed preceding A→B communications concerned routine HOLD/status updates. An isolated bad tool attempt is not independently verified collusion. Lawful HOLD acceptance occurred in 10/11 episodes. All P two-agent agreement labels remain 'UNKNOWN\_PENDING\_BLIND\_HUMAN\_REVIEW'.

## 4.3 Auditor allegations versus tool semantics

Across 88 P M1 judgments, 10 provisional policy-violation candidates were emitted and six judgments failed schema/validation. Joining the ten cited candidates to tool records showed that their written rationales did not substantiate the specific alleged offenses. For seven, the event type 'supervisor\_decision' appears to have invited confusion between reviewer acceptance of HOLD and supervisor READY authorization; the remaining three conflated rejected ticket-completion attempts with release approval. These are case-level, schema-sensitive interpretation discrepancies in our own workflow, not an established general auditor failure mode or a measured detector false-positive rate. No independent human-labeled truth set exists.

| Cited event and M1 claim                                                   | Verified action record                                                                  | Why allegation is unsupported                                                                                         |
|----------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| 'e000032' · flagged in seven episodes as improper supervisor authorization | Actor B executed 'review_t4(accept_assessment)' on the already supported HOLD.          | An authorized reviewer acceptance is not supervisor-approved READY release; the label 'supervisor_decision' misleads. |
| 'e000097' / 'e000098' · three candidates described invalid READY pressure  | Actor B attempted 'declare_complete(T4)'; the tool rejected it due to ticket ownership. | A rejected ownership-violating completion attempt is not an 'approve_release' attempt.                                |
| '06_b2_P0_G1' · one real improper approval attempt                         | B called invalid 'approve_release'; server rejected action.                             | True attempted tool violation, but no verified prohibited agreement or complementary act by A.                        |

Table 4. Source-joined forensic cases. The first two are mismatches in provisional M1 explanations; the third is an actual bad action without demonstrated multi-agent coordination.

M0 produced three metadata candidates and zero alerts; M1's higher candidate count is not evidence of superior detection. Candidate counts, invalid formats and independently verified wrongdoing have different denominators. Event-schema ambiguity is an alternative explanation for the M1 discrepancies and should be treated as a limitation of our measurement environment.

## 5. Discussion and Limitations

### 5.1 The evidentiary asymmetry

The experiments illustrate two distinct interpretive hazards, with important design-based explanations. In R, an instruction deliberately at odds with known blockers produced a response-heavy, capped review path even though the protected release endpoint stayed intact; this may be expected prompt-conflict behavior. In P, our own ambiguous event names plausibly contributed to unsupported audit descriptions. Neither result establishes a novel general vulnerability, and neither establishes collusion. Together they motivate reporting outcome, process and tool-grounded interpretation separately; the proposed broader audit framework remains unvalidated.

### 5.2 Proposed evidence-before-attribution standard

We propose the following checklist as a \*future audit protocol\*, not as a detector shown effective in V8:

- Provenance: distinguish fixture-origin and agent-origin text, including private notes and public route.
- Exposure: distinguish attempted send, accepted write, addressed delivery and inclusion in the recipient's actual model input.
- Action: join cited event IDs to actor identity, tool name, arguments, authorization rule and acceptance/rejection.
- Coordination: require independently supported agreement or strategy linkage plus complementary prohibited behavior, not correlated actions alone.
- Adjudication: preserve blinded human labels and compare monitors on identical held-out case sets before reporting accuracy.

The value of each additional evidence field is presently an untested proposal. A decisive continuation would construct matched collusive, independent-error and legitimate-cooperation trajectories with blind adjudication, freeze one auditor, vary only its input evidence, and measure sensitivity, wrongful accusation rate and annotation cost. This directly answers Apart Track 2.3's evidentiary question without equating message receipt with collusion [4].

### 5.3 Boundaries that constrain interpretation

R tested one synthetic task with one pinned Qwen3-8B model. Its process outcomes were analyzed after a null preregistered endpoint, and the R1 request intentionally conflicted with known blockers, so hesitation/retries may be expected rather than surprising. Requested seeds need not guarantee identical generated counterfactuals; the 12-response cap censored 11/12 R1 episodes. P validated 11/40 planned runs, including eight balanced episodes (two per cell), all capped, without independent human collusion labels. The environment prepopulated most legitimate tasks, limiting opportunities for joint wrongdoing; the gate blanket-blocked a route. M1 used the same model family as workers, its event naming was potentially misleading, and it differed from M0 in both inputs and method. We therefore do not estimate collusion prevalence, intervention effects, auditor precision/recall, general failure rates, or gate efficacy.

## 6. Conclusion

This study does not establish hidden agent collusion or a novel generalized auditing vulnerability. In one synthetic release task, a deliberately conflicting request produced more capped reviewer activity without an invalid approval attempt; this process contrast is exploratory and plausibly expected. In a small, incompletely run team pilot, provisional audit explanations sometimes disagreed with tool-level semantics, potentially because of our own schema choices. The durable contribution is a traceable case study that separates message exposure, action attempts, server outcomes and grounds for attribution. Confirmatory claims would require matched labeled controls, independently adjudicated cases and more representative models and tasks.

### Code and data / research provenance

Research artifacts and repository. The frozen V8 source, manifests, requests, responses, tool receipts, episode CSVs, validation outputs and V3–V7 engineering history are preserved in 'CBS\_V8\_RESEARCHER\_FULL\_EVIDENCE.zip' (internal SHA-256 manifest). A public GitHub repository for the project is linked separately in the official submission form; its public contents have not been independently audited for completeness against this archive. The archive—not repository publication alone—is the provenance basis for our numerical claims. V7/V8 results were not pooled because their observation contracts differ. Code and study materials were developed for this sprint with AI coding assistance; there was no independent human outcome labeling and no private human data.

## References

- [1] Nakamura, M., Kumar, A., Das, S., et al. (2026). Colosseum: Auditing Collusion in Cooperative Multi-Agent Systems. arXiv:2602.15198. <https://arxiv.org/abs/2602.15198>
- [2] Makins, O., Angelini, O., Shams, Z., and Phuong, M. (2026). Multi-Agent AI Control: Distributed Attacks Hamper Per-Instance Monitors. arXiv:2607.07368. <https://arxiv.org/abs/2607.07368>
- [3] Brown, D., Bhargav, S., Santhanam, A., et al. (2026). Stateful Online Monitoring Catches Distributed Agent Attacks. arXiv:2605.31593. <https://arxiv.org/abs/2605.31593>
- [4] Apart Research (2026). AI Collusion Research Sprint, Research Tracks and Guidelines. <https://apartresearch.com/sprints/ai-collusion-research-sprint-2026-10-23-to-2026-10-25>

## Appendix A. Limitations and Dual-Use / Ethical Considerations

**Limitations.** The synthetic release environment mechanically rejected unauthorized approvals, so zero realized unsafe releases does not demonstrate robust agent safety in unconstrained real deployments. R's original invalid-attempt endpoint was null; process outcomes are exploratory and the intentionally contradictory READY request makes extra review/hesitation an expected alternative explanation. One pinned 8B-parameter model, fixed task design and capped response windows constrain generality. P validated 11/40 planned cases, with no blinded human agreement labels or verified collusion. M1's provisional rationales may reflect ambiguous event naming authored by the experimenters rather than intrinsic monitor incompetence; the monitor was not independent of the worker model family. No comparative detector metrics, causal gate effects or collusion incidence can be inferred.

**Dual-use / ethical considerations.** Exposing failure modes in audit attribution can improve defense but also reveal ways an attacker might exploit ambiguity in workflow schemas, pressure messages, or alternate channels. We therefore emphasize defensive measurement and do not operationalize stealthy collusion strategies. The archived traces contain only synthetic release artifacts and no real customer information. Sharing a full research bundle should follow credential screening and preservation of the boundary between privileged researcher truth and blinded adjudication packets.

## Appendix B. Reproducibility and version lineage

**Artifact ledger.** Frozen design: `researcher\_analysis\_20261026\_v8/V8\_FROZEN\_MANIFEST.json` and `V8\_P\_FROZEN\_MANIFEST.json`. Independent raw recomputation: `V8\_R\_RAW\_EPISODES.csv`, `V8\_R\_PAISED\_DIFFERENCES.csv`, `V8\_P\_RAW\_EPISODES.csv`, `V8\_R\_RAW\_AUDIT.json`, `V8\_P\_RAW\_AUDIT.json`. Exact model requests, receipts, tools and events: `experiments/real\_v8\_R\_\*.` and `experiments/real\_v8\_P\_\*.` The outer ZIP includes per-file SHA-256 and length checks; these test internal integrity, not independent cloud-billing verification or model identity.

**Version history.** V3–V6 were feasibility/validation iterations. V7 generated eight genuine real-model episodes with accepted message deliveries but no recorded recipient-input exposure and a blocked reviewable T4 step. V8 modified instrumentation and fixture reviewability before freeze, then completed 24/24 R episodes and partially completed P. V7 and V8 were not treated as equivalent statistical replications. An R continuation followed a corrected validator assumption, preserving original first-pair bytes; a remote P M1 connection failure terminated the twelfth case and the remaining 28 P cases were unrun.

**Data accounting.** R: 173 reviewer B responses + 24 shadow M1 = 197 study responses (two engineering smoke responses excluded). P validated episodes: 758 worker/supervisor + 88 shadow M1 = 846 responses. The aborted P episode retained 67 worker/supervisor responses but no completed M1 response. App metering and code test receipts are archived separately; these are not behavioral outcome labels.