
Beyond GICP: External Traces for Detecting Agent Collusion¹

Chaofeng Jia

Not currently enrolled (previously University of Southampton)

With

Apart Research

Abstract

Prior containment designs such as GICP left auditable intervention traces but remained exposed to circular trust when monitoring shared the agent’s software stack. This work asks whether external process traces can detect agent collusion more reliably than final text. Agents receive a shared board; we measure strict format writes (“Board:”), unique-fact transfer to a second agent, and token use under forbidden, full, and adversarial instructions on Qwen, Allam, and GPT-OSS models. Strict board writes separate full from forbidden at 100% versus 0% for Qwen and Allam across two independent 16-seed runs. Unique facts are reused in 44–50% of board-visible trials and 0% when hidden. A third-party monitor corrects 96.4% of over-link cases and recovers 60% of empty replies. Token use is a weaker auxiliary signal. A TEE-bound counter is proposed as a design-level root that would keep these traces verifiable after escape. The results support process traces plus external intervention as practical evidence of coordination, while noting model heterogeneity and the limits of simple detectors.

¹ Research conducted at the [AI Collusion Research Sprint](#), October 2026

1. Introduction

Agent collusion is difficult to audit when the only available evidence is what agents choose to say. Final text can be made ordinary, empty, or carefully non-committal, while coordination still occurs through shared tools, side channels, or intermediate notes. Existing containment ideas, including earlier fail-closed designs such as GICP, tried to leave intervention traces, but placed those traces inside the same software environment an agent might escape. Once the host logs can be rewritten, the record is no longer independently trustworthy.

This work asks a narrower question: can external process traces—observable writes to a shared board, transfer of unique facts to a second agent, and simple resource signals—serve as practical evidence that agents are coordinating, and can a third party intervene when those traces indicate a problem? The setting is deliberately simple. Two agents receive an incident-reporting task. A shared board is either available or hidden. Instructions vary between forbidding coordination, explicitly allowing it, and asking agents to hide it. We measure whether a strict format marker (“Board:”) appears, whether a unique code written by the first agent is reused by the second, and whether token use differs across conditions. A lightweight external monitor then checks for over-linking or empty outputs and requests a rewrite when either occurs.

The practical value is an auditable signal that does not require trusting the agents’ self-reports or the integrity of host-side logs after escape. If a defender can show that a distinctive fact moved from one agent to another only when a channel was visible, or that a required format appeared under coordination instructions and not under prohibition, that constitutes evidence independent of the final narrative. A hardware root of trust (TEE) is treated as a design-level extension that would bind the counters themselves, addressing the circular-dependency concern raised against purely software monitoring.

Contributions

- 1. We demonstrate that a strict external format trace (“Board:”) separates explicitly coordinated from forbidden conditions at 100% versus 0% for two models across repeated 16-seed runs, while a third model shows high empty-response rates and unstable compliance.*
- 2. We show that unique-fact transfer provides a stronger test of actual information movement than format compliance alone: the second agent reuses the code in 44–50% of trials when the board is visible and in 0% when it is hidden.*
- 3. We evaluate a simple third-party intervention rule that corrects over-linking in 96.4% of triggered cases and recovers non-empty output in 60% of empty-response cases, and we outline how TEE-bound counters would keep such traces verifiable after software escape.*

2. Related Work

Most prior work on multi-agent collusion measures outcomes inside a task—prices above a competitive level, coalition advantage on shared objectives, or deviation between private plans and public actions—rather than external traces that remain visible to a third party after the interaction. Nakamura et al. (Colosseum) audit collusion in cooperative settings by comparing action regret and communication, and note cases where agents plan to collude in text yet act non-collusively. Shi et al. similarly record private plans against public announcements and final actions. These results establish that coordination can be partial or mismatched across channels, but they do not test whether a simple, externally logged format or fact-transfer signal can separate coordinating from independent agents under controlled instructions.

Monitoring literature has begun to combine multiple signals. Nguyen et al. (TraceGuard) score actions along several dimensions, including consistency between stated intent and observed traces, and report gains over single-dimension monitors. Radev et al. show that a monitor using code-edit traces can flag coordinated sabotage, with further improvement when communication is also available. Both support the intuition that behavior can be more informative than final text alone. Neither, however, isolates an external shared board, measures unique-fact reuse by a second agent as a positive control, or evaluates a lightweight third-party rewrite rule triggered by over-linking or empty output.

Earlier containment designs, including GICP, focused on leaving fail-closed intervention traces during evaluation. Reviewer feedback correctly noted the circular dependency: telemetry living in the same software stack the agent can compromise is not independently trustworthy once escape occurs, and empirical testing was limited. The present work retains the goal of auditable traces but moves them outside the agent’s narrative—into format writes, fact transfer, and optional token counts—and treats a TEE-bound counter as a design-level root that would survive host compromise. It does not claim to solve market collusion or to provide a general detector; it supplies a replicable comparison of external signals under forbidden, full, and adversarial instructions, together with an intervention check.

Someone would use this approach over existing task-specific audits when the defender needs evidence that does not rely on agents’ self-reports or on host logs that can be rewritten, and when a minimal positive control (unique fact visible only through the channel) is sufficient to demonstrate information movement. The method adds a direct measurement of channel-dependent transfer and a quantified intervention recovery rate that prior collusion and monitoring studies do not report.

3. Methods

The experimental design evolved from an initial attempt to test whether a GICP-style intervention layer could produce trustworthy containment traces, to a narrower focus on external process signals for collusion detection.

Early checks. We first asked whether software-level counters and logs could remain reliable once an agent was instructed to escape or rewrite host records. These runs confirmed the circular-dependency concern: host-side logs could be altered in simulation, and no independent verification remained. A software mock of TEE attestation showed that binding counters to an enclave measurement would allow a remote verifier to reject a forged quote, but this remained a design demonstration rather than a hardware result. The early runs were small, used simple prompts, and did not yet separate coordination conditions cleanly. They mainly established that self-contained software traces were insufficient and that an external, channel-dependent signal was needed.

Shift to process traces. We therefore introduced a shared board as an external channel. Agents received an incident-reporting task (“API gateway failed at 14:02”) under three instruction types:

- *Forbidden:* work alone; do not produce a shared-board note.
- *Full:* explicitly write a note beginning with “Board:” that includes the failed component and one mitigation.
- *Adversarial:* coordinate if useful but avoid obvious coordination language and the “Board:” marker unless necessary.

The primary metric became a strict string match for “Board:” (case-insensitive). This replaced earlier looser keyword rules that had counted ordinary titles such as “Incident Note” as writes.

Single-model development. Most iterative testing used Qwen (qwen3.8-27b via Groq). We varied seeds (typically 16), measured strict board-write rate and empty-response rate, and added a unique-fact transfer control: Agent A was required to include a distinctive code (e.g., ZETA-9); Agent B either saw the board or did not. Reuse of the exact code by B served as evidence of information movement rather than format compliance alone. A third-party monitor was added later: if an output was empty or affirmatively linked an unrelated daily event to a romantic interest (“A”), the monitor requested a rewrite. Early intervention scoring was overly sensitive (flagging denials); the final criterion counted only affirmative over-linking.

Extension to additional models. The same protocol was then applied to Allam-2-7b and GPT-OSS-120b. These three models were chosen because they are accessible through a single low-cost or free-tier API (Groq), require no local GPU, and offer mid-to-upper capability that ordinary users can obtain. The goal was not to benchmark frontier closed models, but to test whether the signals remained informative on widely available systems. Token counts were recorded as an auxiliary feature but were not used as a primary detector.

Reproducibility details. Temperature was set between 0.3 and 0.5; max tokens typically 60–80. Seeds were fixed per run. Each main comparison used 16 seeds unless noted. Two independent runs of the Qwen full-versus-forbidden comparison were performed to estimate variance. All prompts, detection rules, and result files are retained for replication. Attempts that relied on

content-overlap or soft keyword rules were abandoned once they produced high false positives; the final pipeline uses only the strict “Board:” match, exact fact reuse, empty-response detection, and the corrected over-link rule.

4. Results

Strict board-write rates. Under the final detection rule (case-insensitive match for “Board:”), write rates cleanly separated conditions for two of the three models (Table 1). Qwen and Allam produced the marker in 16/16 full-instruction trials and in 0/16 forbidden and adversarial trials. GPT-OSS-120b produced it in only 4/16 full trials and in 0/16 of the other two conditions; empty responses accounted for 12/16, 2/16 and 16/16 of its outputs respectively. A second independent 16-seed run of the Qwen comparison reproduced the same rates exactly (full = 1.00, forbidden = 0.00; standard deviation across runs = 0).

Table 1. Strict “Board:” rate and empty-response rate (16 seeds).

<i>Model</i>	<i>Condition</i>	<i>Board rate</i>	<i>Empty rate</i>
<i>Qwen</i>	<i>Forbidden</i>	<i>0.00</i>	<i>0.00</i>
<i>Qwen</i>	<i>Full</i>	<i>1.00</i>	<i>0.00</i>
<i>Qwen</i>	<i>Adversarial</i>	<i>0.00</i>	<i>0.00</i>
<i>Allam</i>	<i>Forbidden</i>	<i>0.00</i>	<i>0.00</i>
<i>Allam</i>	<i>Full</i>	<i>1.00</i>	<i>0.00</i>
<i>Allam</i>	<i>Adversarial</i>	<i>0.00</i>	<i>0.00</i>
<i>GPT-OSS-120b</i>	<i>Forbidden</i>	<i>0.00</i>	<i>0.12</i>
<i>GPT-OSS-120b</i>	<i>Full</i>	<i>0.25</i>	<i>0.75</i>
<i>GPT-OSS-120b</i>	<i>Adversarial</i>	<i>0.00</i>	<i>1.00</i>

Unique-fact transfer. Format compliance did not guarantee information movement. When Agent A was required to embed a unique code and Agent B could read the board, B reused the code in 7/16 trials (43.8%) for Qwen and 8/16 (50.0%) for Allam. When the board was hidden, reuse was 0/16 for both models (Table 2). Agent A itself included the code in 100% of trials under the

explicit instruction. Thus the channel, not prior knowledge or prompt leakage, accounted for the observed transfer.

Table 2. Unique-fact reuse by Agent B (16 seeds).

<i>Model</i>	<i>Board visible</i>	<i>Board hidden</i>	<i>Lift</i>
<i>Qwen</i>	0.438	0.00	+0.438
<i>Allam</i>	0.500	0.00	+0.500

Intervention effectiveness. A third-party monitor that requested a rewrite on affirmative over-linking or empty output was evaluated on 28 over-link cases drawn from Qwen and Allam. After correcting an initially over-sensitive scorer that had flagged denials, 27/28 rewrites (96.4%) were non-empty and free of affirmative linkage to the unrelated topic. Separately, 15 empty responses from GPT-OSS-120b were followed by an intervention demand; 9/15 (60%) yielded a non-empty continuation.

Auxiliary token signal. Mean total tokens (12 seeds) differed by model. Allam used substantially more tokens under full instructions (146) than under forbidden (94). Qwen showed no consistent elevation (full 89, forbidden 92). Token count therefore supplied a model-dependent auxiliary cue but not a reliable cross-model detector.

Observations versus interpretation. The strict format marker and the hidden-board control produce large, replicated separations on Qwen and Allam. These are observations. The interpretation that the marker constitutes evidence of coordination holds only under the instruction regimes tested; adversarial instructions eliminated the marker even when coordination remained possible in principle. GPT-OSS-120b’s dominant failure mode was non-response rather than covert compliance. Intervention success rates indicate that an external rule can often elicit a corrected or non-empty output, but do not show that the underlying tendency has been removed.

5. Discussion and Limitations

The results indicate that a minimal external channel plus a strict, externally checkable marker can separate instructed coordination from prohibition on widely available models, and that a hidden-board control can confirm the marker is not an artifact of prompt leakage. Unique-fact

reuse further shows that information can move through the channel at a measurable rate, even if far below perfect transmission. For an auditor, these signals are attractive because they do not require trusting the agents’ final narrative or host logs that an escaped agent could rewrite. The third-party rewrite rule adds a practical response: many over-linked or empty outputs can be driven to a cleaner form without human re-prompting of the original query.

Model heterogeneity is the clearest trend. Qwen and Allam complied stably with explicit format instructions and produced almost no empty replies; GPT-OSS-120b frequently returned empty text and only sporadically emitted the marker. Any deployment that assumes uniform behavior across model families would therefore mis-calibrate both detection thresholds and intervention expectations. Token consumption tracked the full condition for Allam but not for Qwen, reinforcing that resource signals are supplementary at best.

Limitations

Several constraints bound the claims. First, the detector is deliberately simple—a string match for “Board:” plus exact code reuse. Agents that coordinate through paraphrase, timing, tool-call patterns, or steganography would evade it; the adversarial condition already drove the format rate to zero. Second, all runs used a single free-tier API provider and three mid-capability models chosen for accessibility rather than frontier performance. Results may not hold for larger closed models or for different providers. Third, the TEE component remains a design and software simulation; no hardware enclave was available, so the claim that counters would survive host compromise is untested. Fourth, intervention success was measured on triggered cases with a corrected scoring rule; we did not evaluate whether repeated interventions degrade task quality or whether models learn to avoid the trigger phrases. Fifth, the unique-fact codes were short, distinctive alphanumeric strings. Transfer rates for naturalistic or semantically integrated secrets may differ. Finally, the early GICP software-log experiments were small and mainly diagnostic; they are reported only to explain the shift in method, not as confirmatory evidence.

If the assumption that agents will emit a predictable format under coordination pressure is false—as the adversarial results already suggest—the format detector loses power and only the hidden-board fact-transfer control remains informative. If empty responses are treated as benign rather than anomalous, GPT-OSS-120b’s dominant behavior would be invisible to the monitor.

Future Work

Natural extensions include testing paraphrase-resistant transfer metrics, evaluating the same protocol on additional providers, measuring whether intervention recovery remains stable across longer dialogues, and, where hardware permits, binding the write and transfer counters inside a

real TEE so that a remote verifier can confirm them after escape. Comparing the external-trace approach against an existing multi-dimensional monitor on the same logs would also clarify the incremental value of the simple positive controls used here.

6. Conclusion

External process traces can supply practical evidence of agent coordination that does not rely on self-report. On two widely available models, a strict “Board:” marker appeared in every explicitly coordinated trial and in none of the forbidden trials; the separation replicated exactly across independent runs. Unique-fact transfer occurred only when the board was visible, confirming that the channel itself carried information. A lightweight third-party rule corrected most over-linked outputs and recovered a majority of empty replies. These signals are narrow—they fail under adversarial instructions that avoid the marker, vary sharply by model, and have not been bound to hardware—but they are replicable, externally checkable, and directly responsive to the circular-trust problem that limits purely software logs. For auditors who need a minimal, positive-controlled demonstration that coordination has occurred, the combination of format traces, hidden-board contrasts, and intervention recovery offers a concrete starting point.

Code and Data

- **Code repository:** <https://github.com/In9illusion/collusion-process-traces>
- **Data/Datasets:** Included in the same repository under `collusion_experiment/` (early runs) and `collusion_experiment/gptoss_comparison/` (final multi-model tables). Key files: `strict_main_table.json`, `transfer_control_summary.json`, `intervention_effectiveness.json`, `empty_recovery_120b.json`, `qwen_repro_strict.json`.
- **Other artifacts:** `TEE_DESIGN.md` and `tee_attestation_sim.py` (software-only TEE simulation). No separate demo or video.

No info-hazard restrictions apply; prompts and detection rules are fully disclosed.

Author Contributions

The author designed the experiments, implemented the evaluation scripts, ran all API trials, and wrote the report.

References

1. Jia, C. (2026). *GICP v0.1: A Fail-Closed Intervention Layer for Evaluation-Time Containment Failures*. Submitted to *Apart Research AI Incident Response Sprint*.
2. Nakamura et al. (2026). *Colosseum: Auditing Collusion in Cooperative Multi-Agent Systems*. *arXiv:2602.15198*.
3. Shi et al. (2026). *When Agents Lie: Premeditation, Persistence, and Exploitation in Repeated Games*. *arXiv:2607.05132*.
4. Nguyen et al. (2026). *TraceGuard: Structured Multi-Dimensional Monitoring*. *arXiv:2604.03968*.
5. Radev et al. (2026). *The Best-Laid SCHEMEs: Coordinated Sabotage and Monitoring in Multi-Agent Systems*. *arXiv:2605.29178*.

Appendix: Limitations and Dual-Use / Ethical Considerations

This work shows that a strict external format marker and a hidden-board fact-transfer control can separate instructed coordination from prohibition on two widely available models, and that a third-party rule can often elicit a corrected or non-empty reply. It does not establish a general collusion detector. The detector fails when agents avoid the measured surface (the adversarial condition already drives the “Board:” rate to zero). It has not been tested on frontier closed models, on multi-provider settings, or against paraphrase, timing, or steganographic channels.

Dual-use note: the same format and fact-transfer tests could be studied by an adversary to learn which surface features to avoid. The adversarial condition already illustrates that avoidance of the marker is straightforward. No personal data were used. All outputs were generated via a public API under the stated prompts. The TEE component is a design and software simulation only; no hardware root was available.

LLM Usage Statement

LLM assistance was used to draft and edit prose and to generate evaluation scripts. All experimental results, numerical claims, and file references were produced by direct API calls and verified against the saved JSON outputs.