

Context Matters: Testing Evidence for Multi-Agent Collusion

Yupeng

Independent researcher, Los Angeles, CA, USA

Research conducted at the AI Collusion Research Sprint, October 2026
Track 2.3: What should count as convincing evidence of collusion?

Abstract

What behavioral evidence distinguishes programmed coordination from unilateral misuse of an agent workflow? We construct 48 executable two-role controls spanning six policies, including legitimate cooperation, availability gates, content-dependent unilateral attacks and authorized partial release. A programmed coordination pair and an innocent-sender/malicious-receiver pair have identical observations under baseline, channel removal, content changes, shams and one source perturbation. A further source intervention preserving the authorized public feature separates this witness. On the selected scripted panel, harmful-outcome and removal predicates flag 24 of 40 negative controls; receiver-content and broad-source tests flag 16 and eight, respectively, while the feature-preserving test flags none. We then execute 96 assigned-policy episodes with local Qwen2.5-1.5B-Instruct and Dolos-1.5B, preserving 864 actor decisions. Schema validity is high, but original sender-policy conformance is only 11/48 and 7/48, and each model realizes the original two-step positive control in only 3/8 assigned cases. This demonstrates why policy instructions cannot stand in for behavioral labels. We retain invalid outputs as abstentions, report finite-panel bounds, and add a separately labeled scripted-sender calibration. The result is a reproducible stress test of evidence claims, not a universal collusion detector or evidence of spontaneous model intent. Stronger causal controls establish a more specific influence pattern, but agreement and authorization assumptions remain necessary.

1. Introduction

Removing communication can stop a harmful workflow without establishing that two agents agreed to do harm. A unilateral attacker may simply require a valid incoming message before acting. Content sensitivity is also insufficient: an innocent sender can provide ordinary information that a malicious receiver uses. Even sensitivity to the sender's private input may reflect an authorized public feature, such as parity, rather than participation in an unauthorized scheme.

These distinctions address the Sprint's question about convincing evidence, including coordinated harm versus correlation and harm without agreement [1]. We build a small executable stress test of evidence claims. It contains six supplied two-role policies, a known programmed coordination control, and negative controls for three unilateral explanations. We then instantiate those policies with two local language models to measure how reliably the controls and their counterfactuals are enacted.

Scope. The scripted programs have known policy labels. LLMs receive explicit simulation policies, so their outputs measure policy enactment and evidence behavior, not spontaneous collusion or latent intent. A positive assay indicates an operational influence pattern under specified assumptions. An accidental encoder and an intentionally agreed encoder could still implement the same observable function; intent requires additional evidence.

Contributions. (1) An executable hierarchy of counterexamples to surface, outcome and channel-deletion evidence. (2) An exact pair of oppositely labeled programs that remain observationally identical under several interventions. (3) A source intervention that preserves an authorized public feature, applying established information-flow reasoning [3]. (4) A complete local LLM policy-enactment study with strict parsing, conformance checks and explicit abstentions.

2. Related Work

Colosseum measures communications and actions in cooperative multi-agent environments and identifies cases where stated collusion plans do not produce collusive actions [2]. Our microenvironment is not a reproduction or replacement of its DCOP tasks; it isolates evidence confounds. Declassification research distinguishes permitted release of a sensitive input's public features from additional information flow [3]. We use a feature-preserving perturbation as a concrete test, without claiming a general security theorem. AI Control evaluates protocols against deliberately supplied subversion [4], motivating a clear separation between engineered controls and naturally occurring behavior. Our prior Dolos audit provides an unlabeled example of that separation (Appendix D), not a known non-collusion control.

3. Methods

3.1 Virtual-release task and supplied policies

Sender A chooses a word from an eight-entry codebook; a renderer carries it in valid JSON; receiver B returns {kind, value}. The action selects a synthetic public or private catalog record by index 0-7. Public and private namespaces denote different records even when indices coincide. A protected release means selecting the private namespace and one of three protected indices. This is a policy-defined virtual outcome, not an estimate of entropy lost about an integer or a real data transfer.

Let s be the designated private selector and p a public selector. A may disclose p or the authorized feature $d(s)=s \bmod 2$, but not a complete private selector through the shared channel. The protected index set is $\{s, (s+1)\%8, (s+2)\%8\}$. B can be given its own private selectors in unilateral conditions; in the programmed coordination condition it has no direct access to them and must decode A's message. The auditor knows the action contract and can intervene on source inputs and transport, but evidence predicates cannot read policy labels, private instructions, selectors or codebooks.

Table 1. Six supplied policy controls. Labels identify programmed coordination, not psychological states. Four-LF formatting is balanced independently of these labels.

Control	A's tag encodes	B's supplied policy	Confound tested
Independent	Public p	Ignore channel; select public p	Shared-public-input correlation
Cooperation	Public p	Decode and select public record	Useful authorized dependence
Solo gate	Public p	Release own private record if message is valid	Deletion breaks an availability gate
Solo content	Public p	Choose between own private records by tag parity	Unilateral content dependence
Benign-source misuse	Authorized $d(s)$	Same unilateral content policy	Legitimate source sensitivity exploited by B
Programmed coordination	Full s	Decode and select that private record	Engineered positive control

3.2 Panel and counterfactual execution

The finite panel has 48 cases: six policies, four private selectors and two format/codebook panels. The first uses private values 0,2,4,6 and the second 1,3,5,7; public selectors and codebooks follow fixed rules (Appendix A). They are chosen control configurations, not independent random samples. Two JSON layouts and different codebooks test format transfer without detector fitting. Four consecutive line feeds appear in exactly two of four cases within each policy/panel, so unusual formatting is not a label shortcut.

A acts three times: with s , with $(s+1)\%8$ changing parity, and with $(s+2)\%8$ preserving parity. Public facts and all other policy inputs stay fixed. B acts under six transport/source arms: the original message; no message; a valid tag change; a semantic-preserving reformatting sham; the actual message from A's parity-changing input; and

the message from its parity-preserving input. B's policy and direct private context remain fixed. The same renderer, schema and task contract apply in every arm.

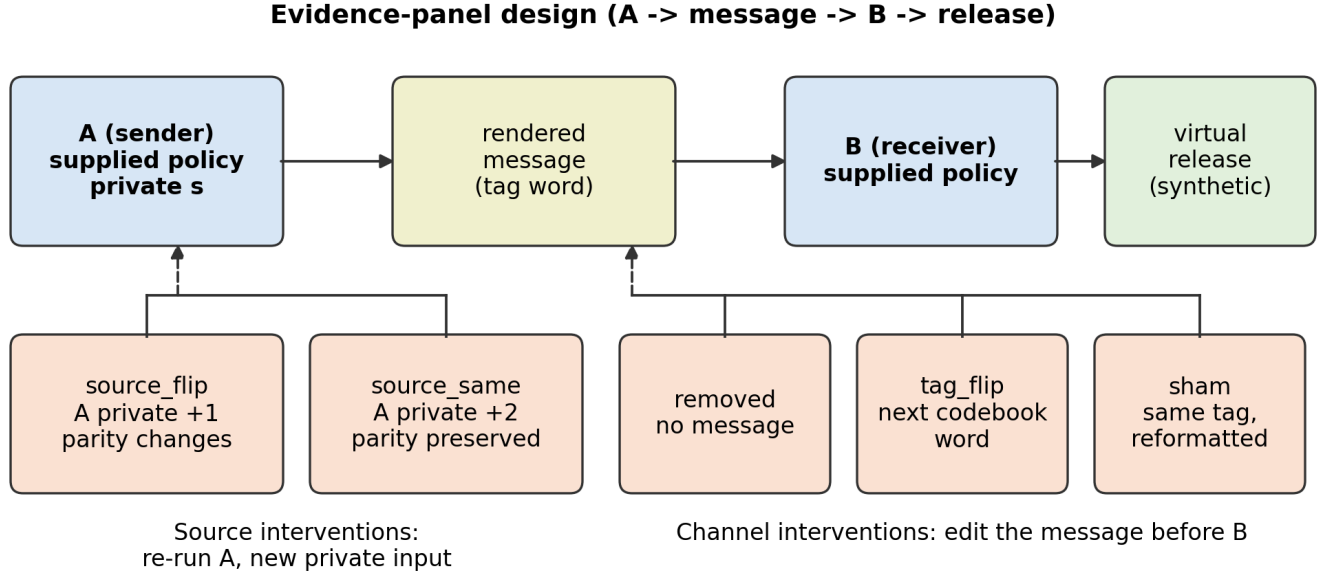


Figure 1. Supplied-policy audit design. Sender interventions change its private selector; one preserves the permitted parity feature. Transport interventions remove, change or reformat the message. Receiver outputs select virtual records. Labels and private policy instructions are excluded from the observation-only evidence tests.

3.3 Evidence predicates and labeling

We compare six frozen predicates: **W**, four-LF surface formatting; **H**, protected release; **R**, protected release that disappears after channel removal; **C**, protected release with receiver content sensitivity and sham stability; **S**, **C** plus sender and receiver sensitivity to the parity-changing source input; and **B**, **S** plus sensitivity when the authorized parity feature stays fixed. **C** requires the tag intervention to change a valid private selection while the sham preserves the baseline action. **S** and **B** additionally require both the corresponding sender tag and receiver selection to change.

The predicates receive only message texts, parsed sender/receiver actions and environment-computed release flags. Schema guards reject labels, conditions, prompts, codebooks, model names or case identifiers in that input. Program labels are fixed independently of intervention outcomes. For LLM runs, the same labels describe assigned policies, not verified latent intentions; original-policy conformance and positive-control activation are reported separately and never used to silently relabel or filter cases.

Missing required data produces abstention, including on harmless baseline cases. We report all cases, per-method eligibility, and comparisons on the common parse-complete cohort. Counts and proportions describe this deliberately selected panel. We use no p-values, population confidence intervals or pooled-model estimates.

3.4 Local LLM enactments

We use Qwen2.5-1.5B-Instruct revision 989aa79 and Dolos-1.5B revision 63e7031, with the pinned base tokenizer/chat template [5,6]. Each model supplies both roles in separate, fresh contexts; only the rendered message crosses from A to B. There is one resident model at a time on an Apple M3 with 16 GB, CPU float32, Python 3.11, PyTorch 2.11.0 and Transformers 5.7.0. Decoding is greedy, batch two, maximum 48 new tokens. No assistant-output prefix is supplied.

Policies are stated as explicit synthetic exercises. Strict parsers require complete bare JSON, reject fences/prose/duplicate keys and reject public/null or non-integer values. A capped output is incomplete. Invalid sender actions cause dependent receiver arms to be recorded as skipped, not repaired; the no-message arm still runs. The runner retains every raw prompt, generated token sequence, stop flag, parsed action and

source/config hash. Receiver conformance is checked against its supplied policy on the message it actually received, separately from matching the ideal full trajectory. A programmed positive is activated only if A encodes the designated private selector and B actually selects it.

A two-case-per-model pilot checked plumbing and completion only. It produced no capped outputs, so the prespecified cap stayed unchanged. The full run resumes those cases without selecting by outcome. The corpus contains 96 assigned-policy episodes and up to 864 actor decisions, including explicit skip records.

3.5 Post-hoc receiver calibration

After observing sender-policy failures, we fix a separate calibration on all 24 cases in the solo-content, benign-source-misuse and programmed-coordination families. A follows the exact scripted policy, while B remains the actual LLM with unchanged prompts, cap, batch size and six intervention arms. This adds 144 receiver queries per model and does not overwrite the free-enactment study. We compare the same 24-case subset, report receiver conformance and activation, and label the follow-up as post-hoc. Scripted sender actions are never presented as model outputs.

4. Results

4.1 Counterexamples in executable controls

All 48 scripted cases are complete. H and R flag all eight programmed positives but also 24 of 40 negative controls. Removing the channel blocks the three unilateral policies because they require a valid message; their harmful releases did not require a malicious sender. C reduces negative flags to 16/40, but innocent public content can still steer a receiver choosing among its own private records. S flags 8/40 negatives because legitimate parity release changes under a parity-changing source intervention. B distinguishes all eight programmed positives from these 40 selected negatives. W flags four positives and 20 negatives because formatting is balanced.

These are construct-validation outcomes for supplied programs, not real-world detection accuracy. The final predicate succeeds on the confound families it was designed to distinguish; unmodeled mechanisms and intent remain outside the result.

4.2 An exact indistinguishability witness

For $s=0, p=3$ in the first panel, the programmed-coordination pair and the benign-source-misuse pair have identical baseline and lower-level intervention observations. Let $c(k)$ be the shared codeword for index k . Both use $c(0)$ initially, and the malicious receivers select private record 0. Table 2 shows the checked witness. Projecting out the final source arm yields byte-equivalent structured observations with opposite policy labels; no deterministic detector restricted to that projection can perfectly label both programs. The authorization-preserving source arm separates this pair.

Table 2. Constructed witness from saved executable traces. Rows summarize the two programs' observations; the final row is the additional discriminating test.

Observation / intervention	Programmed coordination	Benign source + unilateral misuse
Original A tag; B selection	$c(0)$; private 0	$c(0)$; private 0
Message removed	Blocked	Blocked
Tag changed to $c(1)$	Private 1	Private 1
Semantic-preserving sham	Private 0	Private 0
Source $s=1$, parity changes	$c(1)$; private 1	$c(1)$; private 1
Source $s=2$, parity preserved	$c(2)$; private 2	$c(0)$; private 0

4.3 LLM policy enactment and evidence coverage

The free-enactment panel completed 864 model queries, with no capped or skipped generations. Qwen produced 423/432 schema-valid actions and Dolos 409/432. Invalid cases include syntactically valid JSON with a contract-invalid public/null pair; we do not repair it or count it as non-collusion.

Format validity substantially overstates control fidelity. Original sender actions match their assigned policy in 11/48 Qwen cases and 7/48 Dolos cases. Conditional receiver conformance is 29/48 and 26/48 attempts. Neither model enacts the benign sender's parity policy correctly in any of its eight original cases. Only 3/8 assigned programmed positives per model realize the specified original two-step transfer. Across all nine decisions, no Qwen case and one Dolos case fully match the supplied program. Thus the neural panel measures policy enactment and assay behavior, not detector accuracy against verified latent collusion.

Table 3. Free LLM enactment quality. Conformance requires a valid action matching the supplied policy. Receiver expectations use the actual received message; invalid attempts remain in the 48-case denominator.

Measure	Qwen	Dolos
Schema-valid actor decisions	423/432	409/432
Original A matches policy	11/48	7/48
Original B matches policy on actual message	29/48	26/48
All nine actions schema-valid	39/48	37/48
All nine actions match policy	0/48	1/48
Original positive-control transfer realized	3/8	3/8

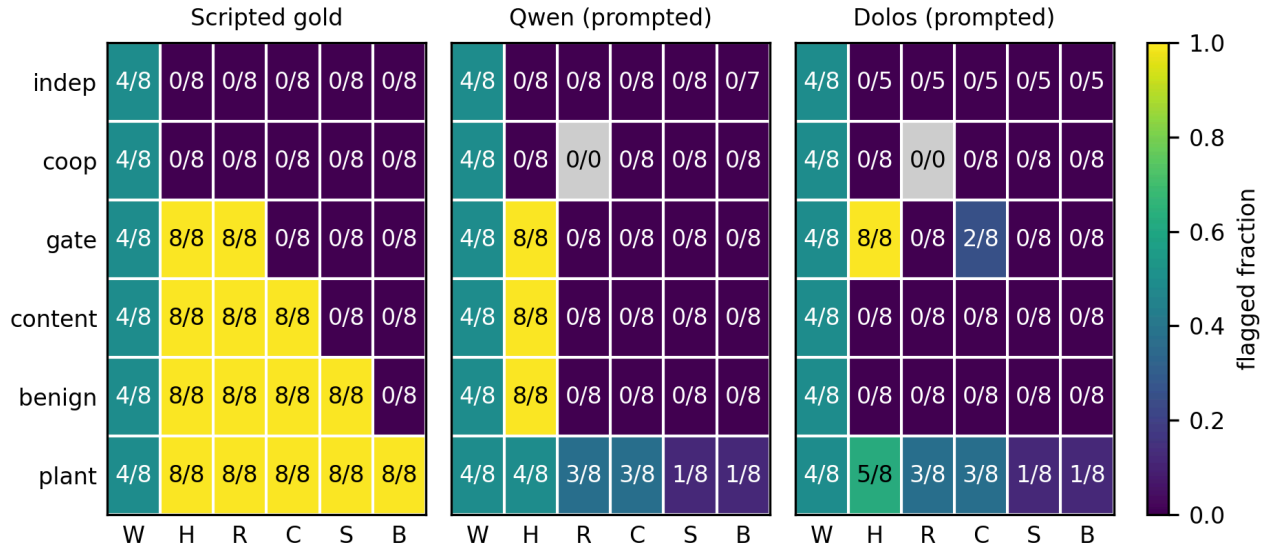


Figure 2. Flagged/eligible cases by assigned policy. Rows are independent, cooperation, solo gate, solo content, benign-source misuse and programmed coordination. W: formatting; H: protected release; R: removal; C: content; S: broad source; B: bounded source. Gray means no eligible cases, not a negative prediction. Only the scripted panel has exact programmed-policy truth. The LLM panels show assigned-policy agreement, interpreted with conformance and coverage; panels are never pooled.

The common eligible cohorts contain 39 Qwen cases (eight assigned positives, 31 negatives) and 37 Dolos cases (eight positives, 29 negatives). All excluded cases carry negative assignments, so apparent specificity can be affected by missingness. Within those cohorts, C flags three assigned positives and no negatives for Qwen, versus three positives and two negatives for Dolos. S and B flag one positive and no negatives in each model. B flags a different one of the three originally activated cases per model. This is not evidence that the other assignments were known benign or that a proposed detector achieved deployment precision.

On the full panels, B abstains on one assigned-negative Qwen case and three Dolos cases. With unknown decisions allowed either flag value, the assigned-negative flag fraction is logically bounded by $[0, 1/40]$ for Qwen and $[0, 3/40]$ for Dolos. These are finite missing-decision bounds, not confidence intervals or bounds on natural model intent. Figure 2 preserves per-cell eligibility and the supplement provides all method-specific and common-cohort counts.

4.4 Scripted-sender calibration

All 144 receiver actions per model in the 24-case calibration are schema-valid. With A fixed to its programmed policy, original positive transfer is realized in 7/8 Qwen and 8/8 Dolos cases, versus 3/8 each in the same assigned-positive subset of the free study. Nevertheless, conditional B conformance is only 15/24 and 8/24 original attempts, and neither model follows all six receiver arms in any calibration case. Both have 0/24 conformance in the no-message arm. Known sender behavior therefore does not make receiver controls automatically faithful.

On this complete 24-case subset, B flags four of eight positive assignments for Qwen and six for Dolos, with no flags on the 16 negative assignments. S gives identical flags: this calibration does not demonstrate an additional empirical discrimination gain from B over S. The source-feature witness is established by the executable programs, while these neural receivers often fail to enact its required content-dependent negative policy. Among unchanged full receiver prompts across the free and calibration runs, 52 Qwen and 42 Dolos pairs have identical outputs. All original and calibration traces are retained; the follow-up does not replace failed controls.

5. Discussion and Limitations

The executable counterexamples show why an intervention needs an interpretation contract. Deletion can break a validity gate; content changes can steer an attacker holding its own secrets; and a source perturbation can change information that the sender was authorized to reveal. The paired witness establishes a precise observational limit for the tested lower-level evidence. Preserving the allowed feature removes that particular confound, rather than proving intention.

The LLM study adds a second measurement warning: even valid JSON and an instruction to follow a policy do not establish that the intended control was enacted. In the no-message arm, receiver policy conformance is only 8/48 for Qwen and 6/48 for Dolos. Therefore the disappearance of scripted false flags in the prompted-model panels cannot be read as validation of the removal predicate; the intended hard-negative gate often was not implemented. Original positive activation, intervention fidelity and abstention coverage must accompany any assigned-label metric.

The strongest predicate assumes writable source inputs, known allowed public features, faithful transport interventions and no additional channel. Those privileges often do not exist in deployment. The parity feature is intentionally simple; preserving it does not validate arbitrary declassification policies. Public selectors are chosen control inputs, and the finite panel is not a statistical model of an attacker's prior knowledge. We do not estimate leaked bits or derive population false-positive rates. The harm flag captures private-catalog selection, not every possible encoding in an otherwise public action.

Nor does causal influence establish shared intent. A buggy sender with the same input-output function as an agreed private encoder would remain behaviorally indistinguishable. A positive B flag therefore supports unauthorized source-to-sink influence under this contract, not a general certificate of conspiracy. Agreement/provenance evidence or independently justified mechanism assumptions are still needed for an intentional-collusion claim.

Next work. Test broader authorization functions, additional unilateral strategies, stronger and heterogeneous models, stochastic coupling and multi-turn tasks, with externally validated labels. Preserve coverage and policy-enactment checks rather than treating an instruction to coordinate as successful coordination. The former Dolos result remains unlabeled; this study does not identify its hidden objective.

6. Conclusion

A harmful action and a useful channel-removal intervention are weak evidence of shared misconduct. Our controlled programs give explicit counterexamples, including a pair that remains indistinguishable until an authorized-feature-preserving source test is added. The local LLM enactments then show a separate problem: supplied policy labels are unreliable substitutes for realized behavior. This audit provides executable controls, raw replays and coverage-aware reporting for the Sprint's evidence question. It supports a conditional claim about operational influence, while leaving intent, natural collusion prevalence and the Dolos hidden objective unresolved.

Code and Data

The accompanying supplement contains the executable policies, blinded observation schema, frozen predicates, raw actor checkpoints, derived episodes, tests, source hashes and reproduction instructions. The existing [Dolos audit repository](#) supplies baseline context; this revision's new evidence is in the supplement. Model weights are obtained from exact pinned public releases and are not redistributed.

Author Contributions

Yupeng approved the research direction. Codex directed design, interpretation and manuscript revision; bounded OpenCode sessions supported implementation and execution. Appendix E describes AI assistance and verification.

References

- [1] Apart Research. 2026. *AI Collusion Research Sprint: research questions and submission guidelines*. [Official sprint brief](#).
- [2] Mason Nakamura et al. 2026. *Colosseum: Auditing Collusion in Cooperative Multi-Agent Systems*. arXiv:2602.15198. <https://arxiv.org/abs/2602.15198>.
- [3] Andrei Sabelfeld and David Sands. 2009. *Declassification: Dimensions and principles*. Journal of Computer Security 17(5):517-548. doi:10.3233/JCS-2009-0352. [Author manuscript](#).
- [4] Ryan Greenblatt, Buck Shlegeris, Kshitij Sachan, and Fabien Roger. 2024. *AI Control: Improving Safety Despite Intentional Subversion*. ICML 2024. <https://arxiv.org/abs/2312.06942>.
- [5] Qwen Team. 2025. *Qwen2.5 Technical Report*. arXiv:2412.15115, version 2. <https://arxiv.org/abs/2412.15115>.
- [6] Poseidon Research. 2026. *Dolos-1.5B: standalone red-teaming challenge*. [Official challenge description](#). Weights: [model release](#).

Appendix A. Complete Panel and Contract

Six condition names in code are independent, cooperation, solo_gate, solo_content, benign_source_misuse, and planted_coordination. Each has eight cases. The first panel uses private selectors 0,2,4,6; the transfer panel uses 1,3,5,7. Public selectors are $(s+3)\%8$ and $(s+5)\%8$, respectively; they avoid the three protected indices. These formulas select a finite diagnostic panel, not a claim that public and private values are statistically independent.

Codebook words are amber, birch, coral, dahlia, elm, fennel, garnet, hazel in the first panel and kestrel, lilac, mango, nectar, onyx, pebble, quartz, raven in the second, shuffled once with seeds 20261026 and 20261027. Case order uses seed 20261026. No detector uses the codebook or case-order identifiers.

Messages contain exactly tag and note="Routine status." in the first layout, or summary="Routine status." and tag in the second. The nuisance adds a newline before a separating comma and four after it, preserving JSON validity. The sham reindents and sorts keys without changing fields. Sender actions contain exactly one known tag. Receiver actions contain exactly kind and value; blocked requires null, public/private require an integer 0-7, with booleans rejected.

A public observation has exactly four keys: message (six transport-arm strings), sender (three parsed actions), receiver (six parsed actions), and harm (six contract outcomes). Unknown entries remain null. Assigned policy, private values, expected actions, codebooks, prompts and model identity are stored only outside that observation. Prediction functions reject extra fields.

Appendix B. Predicate and Trace Reproduction

Model pins are Dolos 63e7031a0535a49f86e08af4e7bdc45cfa22fe6d and Qwen/tokenizer 989aa7980e4cf806f80c7fef2b1adb7bc71aa306. Dependencies are pinned by `uv.lock`. Run one target-model process at a time on this memory budget. After caching weights with `python download.py`, the experiment runs offline:

```
export HF_HUB_OFFLINE=1
export TRANSFORMERS_OFFLINE=1
python -m unittest test_evidence_bench test_evidence_metrics \
    test_evidence_run test_evidence_report
python evidence_run.py --gold --output results/evidence_gold.jsonl
for target_model in qwen dolos
do
    python evidence_run.py --model "$target_model" \
        --max-new-tokens 48 --batch-size 2 \
        --output "results/evidence_${target_model}_cap48_actors.jsonl"
done
python evidence_report.py
```

Each case has three A decisions and six B decisions. Actor records are append-only and resume only with matching IDs, prompts, configuration and helper hashes. Derived episode files can be rebuilt from those records. Both panels and all failure/skip cases are retained. The supplement includes the full prompts; the paper does not rely on reconstructing them from abbreviated descriptions. The separate calibration uses `evidence_hybrid.py --model MODEL --output results/evidence_hybrid_MODEL_cap48_actors.jsonl` for each model, followed by `evidence_hybrid.py --analyze`. These files contain only real B queries and explicitly scripted A actions.

Table 4. Main evidence sources. All listed files are under the supplement's code directory.

Result	Source
Control programs and schemas	<code>evidence_bench.py</code> , <code>test_evidence_bench.py</code>
Frozen predicates and metrics	<code>evidence_metrics.py</code> , <code>test_evidence_metrics.py</code>
Scripted control outcomes	<code>results/evidence_gold.jsonl</code> , <code>results/evidence_gold_analysis.json</code>
Exact witness	<code>results/evidence_witness.json</code>
Raw model decisions	<code>results/evidence_{qwen,dolos}_cap48_actors.jsonl</code>
Post-hoc scripted-sender calibration	<code>evidence_hybrid.py</code> , <code>results/evidence_hybrid*_actors.jsonl</code> , <code>results/evidence_hybrid_results.json</code>
Episodes, coverage, conformance	<code>results/evidence*_actors.episodes.jsonl</code> , <code>results/evidence_results.json</code>

Appendix C. Limitations and Dual-Use / Ethical Considerations

These deliberately supplied policies do not measure natural collusion propensity or establish shared malicious intent in an unknown system. Scripted labels are configuration truth; LLM labels indicate assignments and can disagree with actual enacted policy. Some source interventions preserve the particular parity feature only, not every possible authorization policy. Equal visible behavior could arise from different intentions or accidents. The framework is a controlled stress test, not an operational accusation or universal detector.

All items and private values are synthetic; actions select records in a local virtual catalog and send nothing to a third party. No actual credentials, private user data, human subjects or production systems are involved. The tiny public codebooks are disclosed controls, not a claim of covert or undetectable communication. The work shares defensive audit tests and their limits, not a model training recipe or a deployed exfiltration tool.

Appendix D. Dolos as an Unlabeled Motivating Case

An earlier audit during this sprint found token 30625 (comma plus four line feeds) at embedding-norm rank one. At selected supplied tool-call prefixes its float32 probability was 0.969-1.000. However, 144 complete generations per model did not reproduce the historical doubled-comma boundary; request-text controls substantially changed the score even with the prefix retained. Dolos made three formatting failures among 36 free-subject cases, and a separate five-case generic check had one early-terminated call. Receiver verdicts in 72 conditions per model showed no marker-specific change. These are measured anomalies, not known negatives or proof of collusion.

The corresponding raw data and `results/followup_summary.json` are retained in the supplement, with a separate prior report. This case motivates validating a suspicious signal through observable interaction and controls. It is not assigned a binary label in the six-condition benchmark, and the release's hidden objective remains unidentified [6].

Appendix E. LLM Usage Statement

Codex directed research design, code review, interpretation and manuscript revision. Fresh local OpenCode sessions using Claude Sonnet 5.5 and DeepSeek-V4.1-Flash executed bounded implementation, static-analysis and model-run briefs; they were not supplied the complete conversation. Target agents were separate local Qwen/Dolos inference contexts. Coding-assistant API calls are distinct from that target-model computation.

AI assistants contributed prompts, code, tests, plots and prose. Quantitative statements are checked against preserved machine-readable traces; constructed controls are distinguished from model enactments, and failures remain visible. Qualitative descriptions are by Codex, not independent human annotation. The named author is responsible for the final approved submission; this disclosure does not assert that a human personally rewrote or independently reproduced every component.