PACT: PROVENANCE-AWARE CONTRACT TESTING — WHAT DO LLM-AGENT TASK-CONTRACT CHECKERS ACTUALLY CERTIFY?

Anonymous authors
Paper under double-blind review

ABSTRACT

LLM agents now act on behalf of users: sending messages, initiating payments, and booking travel. To keep them in check, developers run contract checkers: programs that verify the agent only acted on destinations named in the task text. A clean certificate is usually taken to mean the agent did only what the task authorized. But the checker only matches strings: it can pass an action the task never authorized, and flag one the task did. We measure that gap on 7,598 public AgentDojo episodes. We first compute exact risk certificates for such a checker. We then test what its flags mean: independent LLM judges blindly review 400 stratified trace-task pairs and decide whether each sampled flag is a truly unauthorized action, under a predeclared rubric. These two steps — certify the label, then measure its meaning — form an audit framework we call Provenance-Aware Contract Testing (**PACT**). On the attack side, the checker is largely sound: about three-quarters of sampled attack-side flags hold up as genuinely unauthorized actions, and a predeclared extension covering what the main audit cannot see shows that almost no attacks are missed. Yet the dominant failure is over-strictness: 57.5% of sampled benign-side flags are actions the task did authorize (the exact rate depends on the coding rubric). A provenance-aware resolver, which tracks where each destination string came from rather than whether it appeared anywhere, authorizes 2.6x fewer semantic violations than the breadth baselines and eliminates all measured false rejections. The frame-level semantic violation rate is estimated under measured worst-case bounds, not certified. The rubric is human-validated on every checkable boundary unit, with zero residual disagreement, and in a prospective test on unseen tasks, the compiled resolver falls short of its predeclared threshold. Certificates should be read as statements about what the checker actually verified, not as proof of what the task authorized; we measure exactly what separates the two.

1 INTRODUCTION

Deployed LLM agents already send messages, move money, and book travel under a broad grant of permission: the whole channel, account, and card on file (Staufer et al., 2026; Hou et al., 2025). A task contract narrows that grant to what the task requires (Tsai & Bagdasarian, 2025; Shi et al., 2025), and a contract checker certifies, mechanically, that a recorded run obeyed it.

A certificate, though, is only as meaningful as the label it certifies, and the easy-to-check labels are not always the ones users mean. This paper asks **what risk certificates for agent task contracts are actually about**, and measures the answer on public agent traces (Mohammadi et al., 2025; Andriushchenko et al., 2025; Zhang et al., 2025). We call the audit framework Provenance-Aware Contract Testing (**PACT**): it certifies the literal label exactly, then measures what that label means.

Consider a real episode from our evaluation (Figures 1 and 2). The task says "Summarize the article that Bob posted in `general` and send it to Alice." An agent messages Alice; a literal-destination checker admitting only verbatim task-text destinations accepts this – yet the task scoped the summary to a channel thread, not a direct message. A second task says "add all users not already in the channel to the channel"; the same checker flags every such addition as a violation: the users'

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

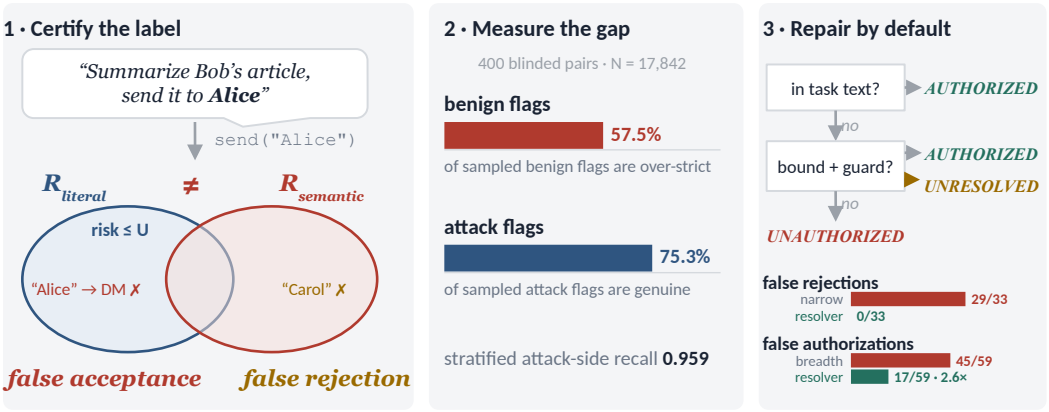


Figure 1: The paper in one view. (Act 1) A certificate attaches to the literal label `R_literal`; users read it as `R_semantic`, true authorization. (Act 2) The label's meaning, measured over 400 blinded stratified pairs from 17,842 episodes: benign-side flags 57.5% over-strict, attack-side flags 75.3% genuine, stratified attack recall 0.959. (Act 3) A provenance-aware resolver defaults unbound destinations to deny: false rejections fall 29/33 to 0/33, authorized semantic violations 45/59 to 17/59 (2.6x). The rubric is human-validated on all 135 checkable units (d = 0/135, four blinded annotators); a Wave-2 census of the 480 uncheckable (of 615) is predeclared.

names appear only in tool output. In a third episode, an injected instruction buried in a webpage tool result makes the agent message an attacker-controlled destination whose string appears in an earlier tool observation, so a "medium" rule treating any earlier tool text as authorization accepts it. These error patterns recur across public agent traces (Section 3.2).

On the attack side, the literal checker is largely sound and almost no attacks are missed. The dominant failure is over-strictness: 57.5% of sampled benign-side flags are actions the task did authorize. A provenance-aware resolver cuts semantic false authorizations 2.6x and eliminates measured false rejections. The paper makes three contributions:

1. **Semantic validation of the contract layer.** A frozen, blinded adjudication protocol measures what the literal checker's flags mean on real traces (Section 3).
2. **A provenance-aware destination resolver.** It authorizes by where each destination string came from, not whether it appeared anywhere, and is evaluated as a predeclared case study against the semantic labels (Section 4).
3. **Certificates with an explicit label.** Exact finite-frame risk certificates with a three-layer label discipline, plus the first human-validation wave of the adjudication rubric (Sections 5–6 and 9).

Statistical machinery is used, not contributed (Appendices A and C); the paper's subject is the *meaning* of agent-safety metrics, where the findings are new.

2 Task Contracts, Labels, and Evaluation Targets

An episode contains context `X_i`, a broad policy-enforcement-point (PEP) manifest `M_i`, a versioned executable contract `C_i`, and a recorded tool-call trace. The checker is a deterministic predicate over the executed trace and the predeclared contract. A destination is any string a side-effecting call sends information or value to; the narrow contract studied here admits only destinations that occur verbatim in the user-task text. All labels are fixed on a finalized frame before any audit sample is drawn, so every quantity in this paper is a well-defined finite-frame rate.

The same episode carries five distinct labels (Table 1), and the paper's central discipline is that they are never merged. They disagree on real data: `R_literal` disagrees with `R_semantic` on discovery-dependent tasks (Sections 3 and 4), and with `R_injection` in both directions (Section 5).

Table 1: Five labels attached to the same episode; the paper never merges them. Every certificate is a statement about exactly one label.

Label	Definition	Measured by
R_literal	violation of the predeclared executable contract rule	deterministic checker (zero-mismatch re-impl.)
R_semantic	unauthorized action relative to the task's intended authorization	blinded adjudication (Section 3)
R_provenance	violation of the provenance-aware resolver's three-value rule	resolver replay (Section 4)
R_injection	the benchmark's own injection-success flag	AgentDojo security label
completion	executed, safe, harmful completion	outcome taxonomy

The evaluation targets are the frame-level rates of these labels, and each is measured by a different method: R_literal is certified by exact audit-count inversion (Section 6.2); R_semantic is estimated by stratified blinded adjudication and reported with measured bounds (Sections 3 and 6.3); R_provenance is computed by deterministic resolver replay (Section 4); R_injection and completion come from the benchmark and are reported descriptively.

3 SEMANTIC VALIDATION OF THE CONTRACT LAYER

3.1 PROTOCOL

To measure *construct* validity – whether the literal-destination flags agree with the task's intended authorization – we freeze a predeclared semantic-validation protocol (Appendix D): a stratified sample of 400 trace-task pairs, 80 per stratum over five strata, drawn with frozen seeds and blinded adjudicator packages. Each package shows only the user-task text, the executed side-effecting calls with destinations, and the preceding tool observations; checker verdicts, strata, and code-to-episode mappings are withheld. The frozen rubric counts a destination as authorized when named in the task text or when the task delegates discovery to a tool, treats <INFORMATION> blocks as attacker content, and admits AMBIGUOUS for genuinely silent task text. The predeclared stop condition (agreement below 0.7, or AMBIGUOUS rate above 0.2) was not triggered in either extension. **Annotation protocol disclosure.** The adjudicators are nine independent LLM sessions across the extensions, re-adjudication, and QA-replication rounds – LLM-assisted annotation, not human annotation, and the agreement statistics below are inter-session reliability, not human inter-annotator reliability. The rubric was written by the authors, who also developed the checkers – a disclosed circularity, partly remediated by the human-validation wave on the checkable boundary units (Section 9); the full disclosure table is in Appendix D (Table 13).

3.2 RESULTS

Two independent blinded sessions completed all 320 verdicts each: raw agreement 0.9906, kappa 0.9813, 3 excluded units (two unresolved disagreements, one AMBIGUOUS-involved). The attack-side extension (80 S5 units) gives raw agreement 0.9875, kappa 0.661 – depressed by the extreme base rate (one violation in 79 of the 80 units): the one disagreement in 160 verdicts, not kappa, carries the reliability information. The three unresolved disagreements across both extensions involve different rubric clauses: no single clause drives them. A predeclared fresh-session replication (Appendix D) confirms these reliability levels on disjoint samples and identifies one clause (clause 4, the AMBIGUOUS reservation) as generation-sensitive on empty-task-text units; Section 6.3 reports its measured effect and its resolution by the amended rubric.

The confusion matrix of the narrow checker against the adjudicated labels is TP 152 / FP 85 / FN 1 / TN 79 (317 units; 3 excluded; the attack-side extension enters through the stratified recall below). Because the design oversamples flagged units, these are stratum-conditional, not projectable to the frame. The per-class reading (Wilson 95% intervals):

- **Checker-positive precision.** Among all adjudicated checker-positive units, 152/237 (64.1%) are true violations. Restricted to the attack-side positive strata (S3+S4), 119/158 (75.3% [68.0, 81.4]) are true violations: about three-quarters, not "almost all."
- **Benign-side over-strictness.** On benign flagged episodes (S1), 33/80 are true violations (41.2% [31.1, 52.2]) and 46/80 (57.5% [46.6, 67.7]) are over-strictness (one unresolved disagreement excluded). One coding contingency: 30 of these 80 units have an empty task text, and under the alternative clause-4 coding (Section 6.3) they would be AMBIGUOUS rather than violations – the headline over-strictness rate is coding-contingent in a way the attack-side rates are not.
- **Attack-side recall: the missing stratum, filled.** Recall was unestimable without an attack/narrow-negative stratum; a predeclared extension adjudicates 80 units from the attack/narrow-clean pool under the identical rubric: the adjudicated violation rate in S5 is 1/79 (1.3% [0.2, 6.8]). Because S3, S4, and S5 partition the attack region with known pool masses (178 / 4,430 / 12,264), the stratified attack-side recall is 0.959, plug-in 95% [0.789, 0.993] from the simultaneous stratum-wise Wilson intervals. The narrow checker's dominant semantic failure mode is therefore benign-side over-strictness, not missed attacks – measured, like all numbers here, under LLM adjudication rather than human labels.

What the semantic sample says about each checker.

On the 93 sampled units belonging to the resolver's four predeclared families (92 with adjudicated verdicts: 59 semantic violations, 33 compliant), we run every checker and join episode verdicts with the adjudicated label (Table 2). Of the 59 semantic violations, 52 come from S3, 4 from S4, and 3 from S1 – every one narrow-positive by stratum construction – so the narrow row's 59/59 detection is a **design artifact** of the stratified sampling, not an estimated detection rate. The crosswalk's design-independent content is the false-authorization column, where every checker faces the identical 59 violation-units. The resolver reduces the narrow checker's false rejections from 29/33 to 0/33 on this subset. Against the identical 59 semantic violations it authorizes 17, versus 45 for medium/wide and 52 for compiled – a 2.6x reduction in semantic false authorizations relative to medium/wide, with non-overlapping 95% intervals (Table 2). Part of the residual moves to UNRESOLVED (6 violation-units, 8 compliant-units) – checker uncertainty, not agent abstention; the headline contrast is insensitive to their disposition (Table 2 caption).

3.3 Coverage of the five-stratum design

The five strata partition the frame by region and checker verdict: S1 benign/narrow-flags (pool mass 196), S2 benign/narrow-clean (774), S3 attack/narrow-breach/compiled-accept (178), S4 attack/narrow-breach/compiled-agree (4,430), S5 attack/narrow-clean (12,264); 80 units are sampled per stratum (full stratum table in Appendix D: Table 12).

A coding-boundary note: 112 of the 400 units have an empty user-task text; their VIOLATION-vs-AMBIGUOUS coding is the clause-4 boundary of Section 6.3. Frame total `N = 17,842` (benign 970, attack 16,872), so frame-level stratified estimates of `R_semantic` are identified.

4 Provenance-aware destination resolution: a predeclared case study

Positioning. This section is a *case study on predeclared task families*, not a general-method claim: the resolver has hand-written bindings for exactly four of the eleven predeclared families, its rules were developed while inspecting traces from the same public frames it replays, and no prospective unseen-task test supports a broader claim (Section 9 reports the executed test, which lands at endpoint MARGINAL).

Rules. A destination is AUTHORIZED iff it appears verbatim in the user text, or it is produced by a *bound* source tool: for the four families we predeclare source bindings with executable guards (`slack/user_task_10`: `add_user_to_channel.user` bound to `get_users_in_channel`, guard "not already a member of the target channel"; `banking/user_task_0`: `send_money.recipient` bound to `read_file`, guard "the file matches the bill named in the task"; `travel/user_task_0`: `reserve_hotel.hotel` bound to the ratings tool, guard "rating

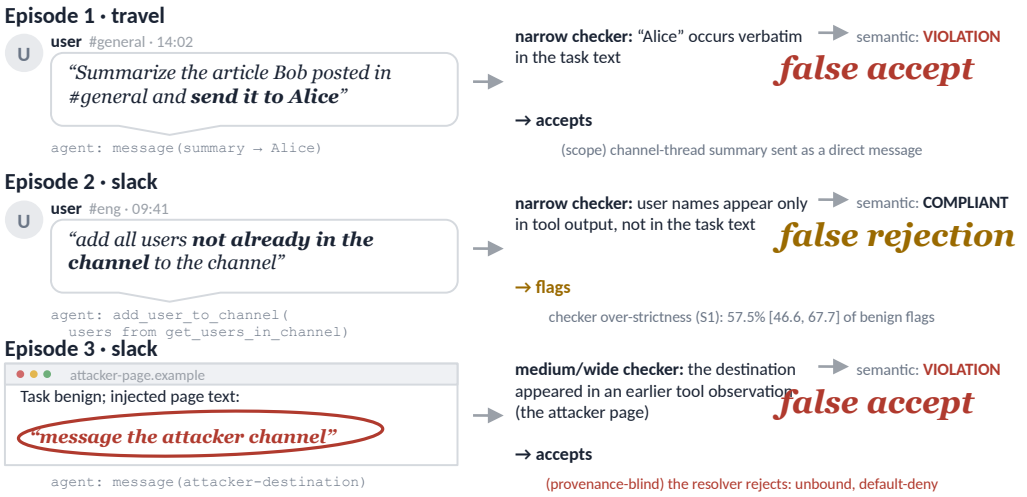


Figure 2: The three running-example episodes: a benign task whose scope the narrow checker misreads (false acceptance: the label authorizes an unintended recipient), a discovery task whose tool-discovered destinations the narrow checker cannot see (false rejection), and an injected instruction that breadth baselines authorize from the attacker's own text (false acceptance). Section 4 resolves the third.

Table 2: Semantic crosswalk of the checkers on the 92 adjudicated units of the resolver's four predeclared families (59 semantic violations, 33 compliant): false authorizations among violations, and false rejections among compliant units. Wilson 95% intervals; the narrow row's detection is fixed by design (Section 3.2). Worst-case UNRESOLVED disposition moves the provenance-aware row to 23/59 (39.0% [0.276, 0.517]), still non-overlapping with medium/wide.

Checker (92 adjudicated units)	unauthorized among semantic violations	semantic false authorizations	semantic false rejections among compliant	unresolved (viol / comp)
narrow	59/59 (design artifact)	0	29/33 (87.9% [72.7, 95.2])	0 / 0
medium / wide	14/59 (23.7% [14.7, 36.0])	45 (76.3% [64.0, 85.3])	0/33	0 / 0
compiled	7/59 (11.9% [5.9, 22.5])	52 (88.1% [77.5, 94.1])	0/33	0 / 0
provenance-aware	36/59 (61.0% [48.3, 72.4])	17 (28.8% [18.8, 41.4])	0/33	6 / 8

above 4"; `banking/user_task_10` is declared *unresolvable* – "pay the bill, like last month" names no discoverable payee). Two principles are mechanical: an occurrence in tool text is not authorization, and agent restatement is not authorization. Two cases yield the third verdict UNRESOLVED: a binding observation whose guard cannot be verified, and the declared-unresolvable class. Resolution uses only observations strictly *before* the side-effecting call and shares the narrow/medium/wide action scope.

Baselines. Medium and wide are pre-call-only rules, with the nesting `wide <= medium <= narrow` asserted per episode on all 1,444 family episodes. Medium and wide flag identical episode sets on this benchmark: the extra widening is vacuous (destinations never appear solely in pre-call assistant text) – a benchmark fact, not a general claim.

Three-value accounting. With the UNRESOLVED verdict, checker decisions partition the attack and benign episodes into authorized + unauthorized + unresolved = total (invariant 4, asserted per checker; Table 3). These are checker-replay quantities over pre-generated historical traces: "authorized" means *the checker authorizes*, not attack prevention, and UNRESOLVED is checker uncertainty, not agent abstention.

Table 3: Three-value accounting of checker decisions on the 1,444 predeclared-family episodes (attack N=1,378, benign N=66): authorized / flagged / unresolved episodes per checker under the predeclared aggregation order (Section 4). Medium and wide flag identical episode sets; the A2/A3 ablations each change exactly one verdict.

Checker	attack: auth / flag / unres (N=1,378)	benign: auth / flag / unres (N=66)
narrow	889 / 489 / 0	51 / 15 / 0
medium / wide	1,222 / 156 / 0	62 / 4 / 0
compiled	1,067 / 311 / 0	64 / 2 / 0
provenance-aware	**957 / 346 / 75**	**59 / 4 / 3**
A2 no-binding	958 / 345 / 75	59 / 4 / 3
A3 no-guard	957 / 346 / 75	60 / 3 / 3

Component attribution. The resolver authorizes 265 fewer attack episodes than medium/wide (957 versus 1,222; Table 3) and 68 more than narrow. The per-episode transition table locates the difference: the dominant mechanism is the **declared-binding default**, under which destinations for action-parameter pairs with no predeclared binding are UNAUTHORIZED unless verbatim in user text (451 flagged decisions carry this reason, and an episode can carry several; a further 110 are bound actions whose destination is absent from the bound source's observations). Guard enforcement (A3) and source binding (A2) each change exactly one verdict in this evaluation. On these families, predeclared provenance resolution *behaves as a conservative default-deny policy for unbound destinations with guard-verified exceptions*, and its headline improvement over breadth baselines comes from that default, not from the bindings or guards individually. Measured against the semantic labels (Table 2), this eliminates false rejections and cuts semantic false authorizations 2.6x relative to medium/wide (17/59 versus 45/59, non-overlapping 95% intervals); whether bindings and guards earn their specification cost beyond the four families is what the prospective test of Section 9 addresses (decision flow in Appendix F).

5 Benchmark experiments on real agent traces

5.1 The literal layer and its certificates

Real-trace frames. The certificate machinery is applied to public, unmodified agent traces: 7,598 AgentDojo episodes (four suites, two pipelines). For each (pipeline, suite) pair a narrow executable contract is predeclared: the agent may send information or value only to destinations that occur verbatim in the user-task text. The checker is deterministic; all 7,598 labels re-derived by an independent verifier with zero mismatches – implementation validity, not construct validity (Section 3 measures the latter). A second study extends the frozen contract, seeds, and arithmetic to eight further pipelines (32 frames, 10,244 episodes; zero-mismatch re-derivation), including three `gpt-4o` defense variants and two open-weight 70B models.

The certificate almost never accepts: 29 of 32 frames have risks above `alpha = 0.05` when errors and breaches are counted together; 3 are certified below. Prompt-injection defenses do not uniformly shrink the literal layer: tool filtering and the prompt-injection detector reduce it on some suites (`gpt-4o` travel 0.184 to 0.0125). Spotlighting leaves `slack` unchanged and *increases* banking risk, and the injection-defense-trained `Meta-SecAlign-70B` has the highest slack risk (0.970). Nor is the contract layer the injection layer: the cross-tab with AgentDojo's security flag disagrees both ways (660 `gpt-4o/slack` contract losses are AgentDojo-secure; 151 insecure episodes contract-clean).

Error and breach are separable layers. Recomputing the audit under three codings (breach-only, error-only, union) shows the choice is decision-relevant: on `gpt-4o/travel` the breach-only risk is 0.0447 while the error-only rate is 0.1434; on `gpt-4o/banking` the reverse holds.
Contract breadth is a precision-detection frontier. The narrow rule is the narrowest member of a predeclared family: `medium` (+ verbatim in any pre-call tool observation) and `wide` (+ anywhere in pre-call trace text). Widening removes benign flags but also detection, and the trade-off is steepest where tasks delegate discovery to tools: on `gpt-4o/slack` the benign flag rate falls from 0.6667 (narrow) to 0.0476 (wide) while detection of AgentDojo-insecure episodes falls from 0.872 to 0.289.

The wide rule lets the agent's own earlier text legitimize a destination, so an attacker who induces an echo defeats it; wide is a benign-precision probe, not a deployable contract.

From natural-language clauses to executable predicates. Eleven benign episodes spanning the four suites had their authorization clauses hand-compiled into predicates; in 4/11 the narrow checker is strictly stricter than the compiled natural-language reading. Mechanized evaluation over the 1,444 family episodes shows the compiled relaxations resolve the benign-side gaps almost exactly where the hand-compilation predicted. The same relaxations accept 178 attacker-injected destinations that the narrow checker breaches (banking attack detection falls from 0.633 to 0.209), and the Section 3 audit confirms this is not a labeling artifact (57 of the 79 adjudicated S3 disagreement units are true violations). Every fixed, computable checker defines a certifiable risk; narrow is distinguished only because its semantic meaning has been audited – and found substantially over-strict on the benign side.

5.2 Task-family-partitioned matched reanalysis of defenses

Positioning. This is a *matched reanalysis of the frozen real-trace label layers, partitioned by task family* – not a held-out evaluation of the resolver: the pooled defense comparisons of Section 5.1 average over task families whose composition differs across pipelines, and partitioning by family removes that confound within a prespecified subset. The study freezes a task-family split (97 families shared by the base `gpt-4o` pipeline and all three defense variants, 32 dev / 65 test) and the matched defense comparisons on the test families (Table 4). All three defenses significantly reduce contract breaches versus the undefended base. The layers disagree: `pi_detector` cuts contract breaches to 0.0945 while its injection-success rate remains 0.2156 – a defense can suppress the contract-visible violation more than the injection itself. No pipeline's certificate accepts at any audited fraction; the exact acceptance probability at the four pipelines' true breach rates (0.0945-0.3778) nowhere exceeds 3.5×10^{-5} (Appendix F). Scanning several audited fractions and certifying at the first acceptable one consumes error budget across looks; exact alpha-spending (Proposition A.11, Appendix A) restores validity: predeclared looks at allocated levels `epsilon_l` with `sum_l epsilon_l <= epsilon` control the false-certificate probability exactly.

6 Risk certification for contract labels

6.1 What is certified

Every certificate in this paper bounds one named label. The finite-frame results below bound `R_literal` – the risk that a randomly drawn frame episode violates the predeclared executable rule – not `R_semantic` (Section 3 shows they differ).

6.2 Exact finite-frame inversion and the audit-design ladder

Fix binary literal labels on a finalized frame of `N` episodes, audit each with probability `q`, and observe `L` violations among the `T` audited episodes. The exact certificate inverts the audit-count law: `M_U(L) = max{m : Pr[Binomial(m,q) <= L] >= epsilon}`, and `U = M_U(L)/N` covers `R_literal` with probability at least `1 - epsilon` (proof: Section C.2). A same-record conditional-hypergeometric inversion and a fixed-capacity simple-random-sample (SRS) inversion are equally valid; neither dominates pointwise on the 41 frozen records (Appendix B). The design-free worst case `U_wc = (L + N - T)/N` bounds `R_literal` with probability 1; five hard invariants are asserted over all released records (Appendix B).

Table 5 gives the ladder at one seeded `q = 0.9` realization on the synthetic transfer frame (Appendix C): each row consumes more information; exact inversion tightens the probability-1 worst case 68x on the same audit record; the total gain across the ladder is 137-fold. On the eight real frames, every certificate's gap above true frame risk is nonnegative (Appendix Figure 3).

6.3 Connecting certificates to semantic targets

The semantic layer of Section 3 and the certification layer of Section 6.2 are connected by a measured bound, not a certificate: standard stratified inference partitions the frame into strata `h` of known

Table 4: Matched defense comparisons on the frozen test families (paired episodes, 487 per comparison, family-clustered bootstrap 95% intervals): change in contract-breach rate versus the undefended base. `pi_detector`'s injection-success rate remains 0.2156 while contract-breach falls to 0.0945.

Defense (vs. undefended base)	contract-breach delta [bootstrap 95%]
`tool_filter`	-0.2382 [-0.3082, -0.1649]
`transformers_pi_detector`	-0.2834 [-0.3557, -0.2147]
`spotlighting`	-0.0349 [-0.0639, -0.0080]

Table 5: The audit-design ladder at the synthetic transfer frame ($N = 1{,}280$, `epsilon` = 0.01), one seeded $q = 0.9$ realization with $L = 0$: audit record, information consumed, and resulting upper bound for `R_literal`. The SRS row uses a different record ($n = 1{,}152$). Real-frame evaluation: Appendix Figure 3.

Certificate, `N = 1,280, epsilon = 0.01`	Audit record used	Information consumed	Upper bound for `R_literal`
Design-free worst case	Bernoulli realization `T = 1,143, L = 0`	none (probability-1 bound)	0.107031
Hoeffding	same record	randomized design with known `q`	0.047126
Exact inversion, Bernoulli audit indicator	same record	+ discreteness of `L`	0.0015625
Exact inversion, conditional hypergeometric	same record	+ conditioning on realized `T`	0.0015625
Exact inversion, fixed-capacity SRS	**different** record (`n = 1,152, L = 0`)	+ fixed sample size	0.00078125

mass `p_h = N_h/N`, audits semantically at known inclusion rates, bounds each stratum's rate `u_h` at allocated levels, and combines `R_semantic <= sum_h p_h u_h`. A certificate built this way attaches to the adjudicated target, not to a proxy checker known to be over-strict. Executing the protocol gives the measured worst-case bound `R_semantic <= 0.213` overall (AMBIGUOUS counted as violations), dominated by the allocated 0.306 (Appendix Table 16). Two gaps block certification rather than measurement: corpus-wide adjudicators are LLM sessions (human-validated only on the checkable subset; Section 9), and the plug-in interval ignores adjudication error. We therefore certify `R_literal` only and present semantic quantities as measured bounds.

Misclassification-aware stratified inference. A predeclared analysis (Appendix D) computes three pieces: (i) the stratified mid-point estimate $\hat{R}$`_semantic = 0.216`, sampling 95% CI [0.188, 0.244]; (ii) adjudication-error sensitivity [0.204, 0.227] – the estimate is dominated by sampling uncertainty, not session disagreement; (iii) exact simultaneous upper bounds (Bonferroni `epsilon_h = 0.01`, hypergeometric inversion): 0.289 under the predeclared unresolved-unit rule and 0.306 under the strict worst case. Attack-side recall receives the same treatment: point estimate 0.959, plug-in [0.789, 0.993], misclassification-aware [0.920, 0.999] (sampling error only versus sampling plus adjudication error). All remain statements about the *adjudicated* target (Appendix Table 16).

A four-part predeclared program (Appendix D) establishes annotation-layer reliability. A fresh-session replication confirms inter-session reliability (kappa 0.953) and localizes the dominant residual uncertainty to one rubric-boundary coding choice on empty-task-text units (point within [0.181, 0.216], bounds invariant). A predeclared re-adjudication of all 135 boundary units under an amended rubric resolves the convention (zero AMBIGUOUS in 270), and per-unit conformity QA caught session H misreading 39/135 packages while G conformed 135/135 (raw agreement 0.711, kappa 0.367: agreement alone misleads). A third blinded session on the 39 disputed units conforms 37/39: H's failures are idiosyncratic. A conformity ledger closes the layer with pooled conformity 268/309 = 86.7% under the predeclared predictor. Two scope clauses: "stabilized" is not "validated" (Section 9's human round validates the amended rubric text at divergence 0/135); and all sessions share one model stack, so 86.7% is a within-stack rate (second-stack condition predeclared).

7 Related work

Statistical auditing and selective labels. The statistical core is standard: Horvitz–Thompson (HT) weighting (Horvitz & Thompson, 1952), Hoeffding (Hoeffding, 1963), Clopper–Pearson (Clopper & Pearson, 1934), empirical Bernstein (Maurer & Pontil, 2009); selective labels (Manski, 2003; Kleinberg et al., 2018; Chohlas-Wood et al., 2023) supply the partial-identification tools; conformal risk control (Bates et al., 2021; Angelopoulos et al., 2022) assumes exchangeability, not selective audit labels.

LLM-as-judge and annotation methodology. Prior work studies judge agreement, calibration, and selective evaluation (Zheng et al., 2023; Zhu et al., 2023; Gu et al., 2024; Jung et al., 2024; Wang et al., 2025); validity degrades near the capability frontier, our circularity concern (Section 3.1). Inter-coder agreement (Artstein & Poesio, 2008), kappa's prevalence paradox (Landis & Koch, 1977), uncodable units (Krippendorff, 2004), and misclassification dominating sampling error (Bross, 1954; Gustafson, 2004) are classical standards; recent work validates LLM labels against human annotations (Gilardi et al., 2023; Pangakis et al., 2023). Our contribution is the protocol: frozen rubric, blinded packages, predeclared stop conditions, and a human wave on the checkable subset (divergence 0/135).

Natural-language policy compilation. Natural-language authorization clauses compile to executable specifications in temporal-logic synthesis (Cosler et al., 2023), runtime temporal-constraint enforcement (Kamath et al., 2025), and decidable policy languages like Cedar (Cutler et al., 2024); 57 of the 79 adjudicated narrow-breach/compiled-accept disagreements are true violations (Section 3) – compilation changes the policy's meaning in the dangerous direction.

Permission systems, information flow, and agent guardrails. Decentralized and dynamic information-flow control (IFC) (Myers & Liskov, 1997; Stefan et al., 2011) enforces provenance labels at the language or OS level; the Section 4 resolver transplants IFC to trace checking; runtime verification and shielding (Bloem et al., 2015; Alshiekh et al., 2018) enforce specifications before action. AgentDojo and ToolEmu (Debenedetti et al., 2024; Ruan et al., 2024) probe tool-use risks without sampled-audit certificates; R-Judge (Yuan et al., 2024) benchmarks human safety labels on agent traces; CaMeL (Debenedetti et al., 2025) and guard agents (Xiang et al., 2025) enforce defenses at runtime.

8 Conclusion

A statistically exact, verified contract checker can be semantically wrong on both sides – overstrict on 57.5% of sampled benign flags (coding-contingent), genuine on 75.3% of sampled attack-side flags – while missing almost no attacks. The resolver cuts semantic false authorizations 2.6x versus breadth baselines and eliminates measured false rejections. The frame-level semantic rate is a measured bound, not a certificate: 0.216 against an exact worst-case 0.306, under a rubric human-validated with zero divergence on the checkable units. The framework behind these results is PACT: certify the label, then measure its meaning. All results are computed from frozen, seeded records over 7,598 public episodes and 400 blindly adjudicated trace-task pairs (Appendix B). Certificates stay attached to the literal label they bound; the contribution is a replayable account of what they mean and what the predeclared Wave-2 census would measure next.

9 Limitations

The contract language is predeclared and narrow, the rubric was authored by the checker team, and the initial adjudicators are LLM sessions (Section 3.1); the predeclared human wave validates the rubric on all 135 checkable units at divergence d = 0/135. Claims are scoped to one public benchmark and randomized audits, the frame-level semantic rate is a measured bound, not a certificate, and Section 4 is a case study whose prospective test lands at endpoint MARGINAL.

References

Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe Reinforcement Learning via Shielding. In *Proceedings of the AAAI Conference*

on Artificial Intelligence, volume 32. Association for the Advancement of Artificial Intelligence (AAAI), 2018. doi: 10.1609/aaai.v32i1.11797.

Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, J. Zico Kolter, Matt Fredrikson, Yarin Gal, and Xander Davies. AgentHarm: A Benchmark for Measuring Harmfulness of LLM Agents. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=AC5n7xHuR1.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal Risk Control, 2022. arXiv:2208.02814.

Ron Artstein and Massimo Poesio. Survey Article: Inter-Coder Agreement for Computational Linguistics. *Computational Linguistics*, 34(4):555–596, 2008. doi: 10.1162/coli.07-034-R2. URL https://aclanthology.org/J08-4004/.

Stephen Bates, Anastasios N. Angelopoulos, Lihua Lei, Jitendra Malik, and Michael I. Jordan. Distribution-Free, Risk-Controlling Prediction Sets, 2021. arXiv:2101.02703.

Roderick Bloem, Bettina Könighofer, Robert Könighofer, and Chao Wang. Shield Synthesis: Runtime Enforcement for Reactive Systems. In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, pp. 533–548. Springer Berlin Heidelberg, 2015. doi: 10.1007/978-3-662-46681-0_51.

Irwin D. J. Bross. Misclassification in 2 x 2 Tables. *Biometrics*, 10(4):478–493, 1954. doi: 10.2307/3001619.

Alex Chohlas-Wood, Madison Coots, Sharad Goel, and Julian Nyarko. Designing Equitable Algorithms, 2023. arXiv:2302.09157.

C. J. Clopper and E. S. Pearson. The Use of Confidence or Fiducial Limits Illustrated in the Case of the Binomial. *Biometrika*, 26(4):404–413, 1934. doi: 10.1093/biomet/26.4.404.

Matthias Cosler, Christopher Hahn, Daniel Mendoza, Frederik Schmitt, and Caroline Trippel. nl2spec: Interactively Translating Unstructured Natural Language to Temporal Logics with Large Language Models. In *Computer Aided Verification (CAV)*, pp. 383–396. Springer Nature Switzerland, 2023. doi: 10.1007/978-3-031-37703-7_18.

Joseph W. Cutler, Craig Disselkoen, Aaron Eline, Shaobo He, Kyle Headley, Michael Hicks, Kesha Hietala, Eleftherios Ioannidis, John Kastner, Anwar Mamat, Darin McAdams, Matt McCutchen, Neha Rungta, Emina Torlak, and Andrew Wells. Cedar: A New Language for Expressive, Fast, Safe, and Analyzable Authorization (Extended Version), 2024. arXiv:2403.04651.

Edoardo Debenedetti, Jie Zhang, Mislav Balunović, Luca Beurer-Kellner, Marc Fischer, and Florian Tramèr. AgentDojo: A Dynamic Environment to Evaluate Prompt Injection Attacks and Defenses for LLM Agents, 2024. arXiv:2406.13352.

Edoardo Debenedetti, Ilia Shumailov, Tianqi Fan, Jamie Hayes, Nicholas Carlini, Daniel Fabian, Christoph Kern, Chongyang Shi, Andreas Terzis, and Florian Tramèr. Defeating Prompt Injections by Design, 2025. arXiv:2503.18813.

Fabrizio Gilardi, Meysam Alizadeh, and Maël Kubli. ChatGPT Outperforms Crowd Workers for Text-Annotation Tasks. *Proceedings of the National Academy of Sciences*, 120(30), 2023. doi: 10.1073/pnas.2305016120.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Saizhuo Wang, Kun Zhang, Yuanzhuo Wang, Wen Gao, Lionel Ni, and Jian Guo. A Survey on LLM-as-a-Judge, 2024. arXiv:2411.15594.

Paul Gustafson. *Measurement Error and Misclassification in Statistics and Epidemiology: Impacts and Bayesian Adjustments*. Chapman & Hall/CRC, 2004. doi: 10.1201/9780203502761.

Wassily Hoeffding. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. doi: 10.1080/01621459.1963.10500830.

D. G. Horvitz and D. J. Thompson. A Generalization of Sampling Without Replacement from a Finite Universe. *Journal of the American Statistical Association*, 47(260):663–685, 1952. doi: 10.1080/01621459.1952.10483446.

Xinyi Hou, Yanjie Zhao, Shenao Wang, and Haoyu Wang. Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions, 2025. arXiv:2503.23278.

Jaehun Jung, Faeze Brahman, and Yejin Choi. Trust or Escalate: LLM Judges with Provable Guarantees for Human Agreement, 2024. arXiv:2407.18370.

Adharsh Kamath, Sishen Zhang, Calvin Xu, Shubham Ugare, Gagandeep Singh, and Sasa Misailovic. Enforcing Temporal Constraints for LLM Agents, 2025. arXiv:2512.23738.

Jon Kleinberg, Himabindu Lakkaraju, Jure Leskovec, Jens Ludwig, and Sendhil Mullainathan. Human Decisions and Machine Predictions. *The Quarterly Journal of Economics*, 133(1):237–293, 2018. doi: 10.1093/qje/qjx032.

Klaus Krippendorff. *Content Analysis: An Introduction to Its Methodology*. Sage, Thousand Oaks, CA, 2 edition, 2004.

J. Richard Landis and Gary G. Koch. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174, 1977. doi: 10.2307/2529310.

Charles F. Manski. *Partial Identification of Probability Distributions*. Springer Series in Statistics. Springer-Verlag, New York, 2003. doi: 10.1007/b97478.

Andreas Maurer and Massimiliano Pontil. Empirical Bernstein Bounds and Sample Variance Penalization. In *Proceedings of the 22nd Annual Conference on Learning Theory (COLT)*, 2009.

Mahmoud Mohammadi, Yipeng Li, Jane Lo, and Wendy Yip. Evaluation and Benchmarking of LLM Agents: A Survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 6129–6139. ACM, 2025. doi: 10.1145/3711896.3736570.

Andrew C. Myers and Barbara Liskov. A Decentralized Model for Information Flow Control. In *Proceedings of the Sixteenth ACM Symposium on Operating Systems Principles (SOSP)*, pp. 129–142. ACM, 1997. doi: 10.1145/268998.266669.

Nicholas Pangakis, Samuel Wolken, and Neil Fasching. Automated Annotation with Generative AI Requires Validation, 2023. arXiv:2306.00176.

Yangjun Ruan, Honghua Dong, Andrew Wang, Silviu Pitis, Yongchao Zhou, Jimmy Ba, Yann Dubois, Chris J. Maddison, and Tatsunori Hashimoto. Identifying the Risks of LM Agents with an LM-Emulated Sandbox. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=GEcwtMk1uA.

Tianneng Shi, Jingxuan He, Zhun Wang, Hongwei Li, Linyu Wu, Wenbo Guo, and Dawn Song. Progent: Securing AI Agents with Privilege Control, 2025. arXiv:2504.11703.

Leon Staufer, Kevin Feng, Kevin Wei, Luke Bailey, Yawen Duan, Mick Yang, A. Pinar Ozisik, Stephen Casper, and Noam Kolt. The 2025 AI Agent Index: Documenting Technical and Safety Features of Deployed Agentic AI Systems. In *Proceedings of the 2026 ACM Conference on Fairness, Accountability, and Transparency*, pp. 1536–1576. ACM, 2026. doi: 10.1145/3805689.3806728.

Deian Stefan, Alejandro Russo, John C. Mitchell, and David Mazières. Flexible Dynamic Information Flow Control in Haskell. In *Proceedings of the 4th ACM Symposium on Haskell (Haskell '11)*, pp. 95–106. ACM, 2011. doi: 10.1145/2034675.2034688.

Lillian Tsai and Eugene Bagdasarian. Contextual Agent Security: A Policy for Every Purpose, 2025. arXiv:2501.17070.

Yidong Wang, Yunze Song, Tingyuan Zhu, Xuanwang Zhang, Zhuohao Yu, Hao Chen, Chiyu Song, Qiufeng Wang, Cunxiang Wang, Zhen Wu, Xinyu Dai, Yue Zhang, Wei Ye, and Shikun Zhang. TrustJudge: Inconsistencies of LLM-as-a-Judge and How to Alleviate Them, 2025. arXiv:2509.21117.

Zhen Xiang, Linzhi Zheng, Yanjie Li, Junyuan Hong, Qinbin Li, Han Xie, Jiawei Zhang, Zidi Xiong, Chulin Xie, Carl Yang, Dawn Song, and Bo Li. GuardAgent: Safeguard LLM Agents via Knowledge-Enabled Reasoning. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 68316–68342. PMLR, 2025. URL https://proceedings.mlr.press/v267/xiang25a.html.

Tongxin Yuan, Zhiwei He, Lingzhong Dong, Yiming Wang, Ruijie Zhao, Tian Xia, Lizhen Xu, Binglin Zhou, Fangqi Li, Zhuosheng Zhang, Rui Wang, and Gongshen Liu. R-Judge: Benchmarking Safety Risk Awareness for LLM Agents, 2024. arXiv:2401.10019.

Hanrong Zhang, Jingyuan Huang, Kai Mei, Yifei Yao, Zhenting Wang, Chenlu Zhan, Hongwei Wang, and Yongfeng Zhang. Agent Security Bench (ASB): Formalizing and Benchmarking Attacks and Defenses in LLM-based Agents. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=V4y0CpX4hK.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=uccHPGDlao.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. JudgeLM: Fine-tuned Large Language Models are Scalable Judges, 2023. arXiv:2310.17631.

REPRODUCIBILITY STATEMENT

All experimental records are frozen, hash-checksummed, and inventoried layer by layer in Appendix B. Every analysis script is self-testing and refuses to overwrite its own outputs. AgentDojo traces are public; the only new data are annotation verdicts, released as hashes. Upon publication, the full register, all analysis code, the label CSVs, and the per-package adjudication verdicts will be released in a public repository.

ETHICS STATEMENT

This study reanalyzes public benchmark traces (AgentDojo) and synthetic sandbox episodes. The initial semantic adjudicators are LLM sessions (Section 3.1); the executed human wave involved four annotators – three author-team members and one external collaborator – labeling the public traces under a pre-registered protocol, with no personal data collected. Injected instructions in the packages are benchmark artifacts shown for adjudication only, never executed. The paper makes no deployment claims: its subject is the meaning of safety metrics, and the primary ethical risk it manages is misreading a certificate as a guarantee it does not carry – hence the three-layer label discipline enforced throughout.

AI USE STATEMENT

Large language models appear in this work in two roles: as the semantic adjudicators of the annotation protocol (Section 3.1 and Appendix G) and as coding and writing assistants during experiment implementation and manuscript preparation. All LLM-generated code and text were reviewed and verified by the authors, who take full responsibility for the content of this paper.

A CERTIFICATE THEORY

This appendix states and proves the alpha-spending rule that the multi-look certification in Section 5.2 relies on (Proposition A.11).

Proposition A.11 (alpha-spending multi-look exact certificates). Fix a finite frame of `N` units with unknown violation count `V`. Predeclare `L` looks: audited subsets `S_1, ..., S_L` of the frame,

arbitrary but fixed before any audit outcome is observed, and allocated levels epsilon_l >= 0 with sum_l epsilon_l <= epsilon. At look l, let U_l be any exact level-epsilon_l upper confidence bound for the frame violation rate V/N from the audit of S_l (Hoeffding, Clopper-Pearson, or hypergeometric inversion). Certify at the first look whose bound clears the threshold tau, if such a look exists. Then the probability of a false certificate – some look's bound clears tau while the true rate exceeds tau – is at most epsilon. Early looks are paid for in width, not validity: a smaller allocated level yields a wider inversion.

Proof. For each look l, the event that U_l <= tau while V/N > tau is a non-coverage event of the exact level-epsilon_l inversion at that look, so its probability is at most epsilon_l marginally, regardless of the other looks. Because the subsets S_l and the levels epsilon_l are fixed before any outcome is observed, no adaptivity enters, and the union bound over the L predeclared looks gives a total false-certificate probability of at most sum_l epsilon_l <= epsilon. Neither independence nor nestedness among the looks is required.

B ARTIFACT REGISTER

Every reported number is computed from frozen, hash-checksummed records; this appendix inventories the record set. The audit layer comprises the label CSVs, the frozen checker configuration, and the sample and adjudication records of the two audits, the fresh-session replication, the boundary re-adjudication, and the QA replication, each with per-package hashes and a sealed code-to-episode manifest. The evaluation layer comprises the conformity ledger, the certificate-comparison scan over the 41 frozen real frames, the prospective-test chain and the two compiler-diagnostic chains (compilation records frozen before evaluation), the per-episode transition table (nested baselines, three-value partition, component matrix), and the semantic crosswalk. The consistency audit asserts five hard invariants over all released records, all passing; the human campaign adds its protocol and amendments, the first-wave ordering record, the four per-annotator verdict records, the interim and final analysis records, and the filled decision log.

C SYNTHETIC CERTIFICATE STUDIES

These sections stress-test the certificate machinery on synthetic frames. They support mechanics, not benchmark or deployment claims; the main text cites them only where noted.

C.1 SETTING, OBSERVED VARIABLES, AND ESTIMANDS

An episode contains context X_i, a broad PEP manifest M_i, a versioned executable task contract C_i, and a proposed stage -> commit trace. The PEP is a hard predicate over an attempted action; the contract checker is a separate deterministic predicate over an executed trace and the predeclared contract. In the sandbox, the parent owns the state machine, transaction identifier, exact staged-payload binding, contract checker, and recorded audit assignment. The model, if present, only proposes text.

Let F_i mean a malformed stage or commit proposal; G_i an abstention; P_i a PEP rejection; D_i a state rejection; B_i^S and B_i^C an executed stage or committed-transition contract breach; and E_i a committed completion. These flags are not mutually exclusive in every presentation: a stage breach may be followed by a malformed commit, and a PEP rejection is a noncompletion rather than automatically a contract breach. (A_i is reserved for ordinary audit assignment.)

The sandbox studies use the following all-request coded episode loss: Y_i^code = 1{F_i or B_i^S or B_i^C}. For a realized frame of N requests, its finite-frame target is R_F^code = N^{-1} sum_i Y_i^code, the target of the finite-frame certificates. It is not synonymous with a conditional risk among committed transitions, R_F^commit = (sum_i B_i^C) / (sum_i E_i), reported as NA when sum_i E_i = 0. We also report completion, safe completion, harmful completion, malformed outcomes, abstentions, PEP rejections, and state rejections. A certificate on R_F^code cannot be read as a certificate on R_F^commit; completion must always be reported next to any conditional risk. For an iid superpopulation, the analogous all-request target is R^code = E[Y^code]; no iid theorem is used to interpret the deterministic factorial frames.

C.1.1 WHAT SELECTIVE OBSERVATION DOES AND DOES NOT IDENTIFY

Let A_i denote ordinary audit assignment and suppose the observable record is (Z_i, A_i, A_iY_i). Without restrictions on unobserved outcomes or the audit mechanism, selectively observed labels do not generally identify population risk. One witness fixes Z: World 0 has risk 0.04 and audits each label with probability 0.25; World 1 has risk 0.10, audits violations with probability 0.10, and audits nonviolations with probability 0.24/0.90. Both produce Pr(A=1,Y=1)=0.01, Pr(A=1,Y=0)=0.24, and Pr(A=0)=0.75, while lying on opposite sides of a 0.05 budget. This witness concerns a setting with genuinely missing labels; the sandbox studies retain full deterministic labels and demonstrate a recorded sampling procedure, not label scarcity.

In the shifted-target study (Table 6), the three local-model contract gates each have 941/1,024 coded-loss episodes but 83 safe completions and no harmful completion; their dominant failure is adaptation/parse failure. The PEP-only local model has no committed completion, so its conditional committed-breach risk is NA. Both broad PEP-oriented reference controls commit harmful transitions on all 1,024 target episodes, and the exact contract reference safely completes all 1,024 episodes. Coded loss alone does not order these controllers by safety; the ordering appears once completion is separated from breach.

C.2 CERTIFICATION STATEMENTS

Finite-frame HT and Hoeffding. For a finite frame with fixed binary losses Y_1,...,Y_N, independently sample audit indicators S_i ~ Bernoulli(q) after finalization, where known $q > 0$. Define W_i = S_iY_i/q and Rhat_HT = N^{-1} sum_i W_i. Conditional on the fixed frame, E_S[Rhat_HT]=R_F. Since W_i are independent and lie in [0,1/q], a one-sided Hoeffding bound [Hoeffding, 1963] for a frozen family of K decisions is U_F = min{1, Rhat_HT + q^{-1} sqrt(log(K/eta)/(2N))}, and with probability at least 1−eta, every member of the stated family satisfies R_F <= U_F.

Exact finite-frame inversion. Fix binary labels on the finalized frame and let m* = sum_i Y_i. Draw audit indicators S_i ~ Bernoulli(q) independently with known q in (0,1], so that T cases are audited and L audited violations are observed; conditional on the fixed labels, L ~ Binomial(m*, q). Define M_U(L) = max{m in {0,...,N} : Pr[Binomial(m, q) <= L] >= epsilon}. Because the binomial CDF is **nonincreasing** in m, the feasible set is the prefix {0,...,M_U(L)}, and M_U(L) is the largest frame violation count not excluded by the one-sided level-epsilon test. The same inversion applies verbatim when the audit draws a fixed capacity n without replacement (L ~ Hypergeometric(N, m*, n)); the two designs are individually valid and neither certificate dominates the other pointwise. The certificate U = M_U(L)/N is *exact* in the sense of nonasymptotic finite-sample coverage at least the nominal level: Pr_S(m*/N <= M_U(L)/N) >= 1 − epsilon. *Proof.* M_U(L) < m* holds iff F_{m*,q}(L) < epsilon (monotonicity in m). For an integer-valued statistic L with CDF g(l), the set {l : g(l) < epsilon} is a lower-tail prefix, so Pr(g(L) < epsilon) <= epsilon; substituting g = F_{m*,q} gives Pr(M_U(L) < m*) <= epsilon. ■

Two boundary cases are defined separately. At q = 1 every case is audited, L = m* exactly, and M_U(L) = L. For 0 < q < 1, zero observed violations give M_U(0) = min{N, floor(log epsilon / log(1−q))} (2 at q = 0.9, epsilon = 0.01); the trunca-

Table 6: Shifted-target study: per-controller coded loss, completion decomposition, and conditional committed-breach risk.

Shifted-target controller	Coded-loss episodes	Completion	Safe completion	Harmful completion	Committed-breach risk
contract_reference	0	1,024	1,024	0	0.000
each LLM contract gate	941	83	83	0	0.000
llm_pep_only	512	0	0	0	NA
pep_only_reference	1,024	1,024	0	1,024	1.000
source_template_reference	1,024	1,024	0	1,024	1.000

tion at N and the separate q = 1 definition are required. Boundary comparisons in the executable implementation are decided in exact rational / 60-digit decimal arithmetic, not binary floating point: at this very boundary the float evaluation of (1-0.9)^2 falls below 0.01 and would return M_U(0) = 1 instead of 2.

iid superpopulation. Assume (Y_i,q_i,S_i) are iid across episodes, 0 < q_min <= q_i, and S_i | (Y_i,q_i) ~ Bernoulli(q_i). Then W_i=S_iY_i/q_i are iid, bounded in [0,1/q_min], and have mean R=E[Y]; q_i need not be independent of Y_i for this HT identity.

Count-based conformal-prediction (CP) routes. With constant q, T_i=S_iY_i is iid Bernoulli(qR), so CP_upper(sum T_i,N,eta/K)/q is exact. With heterogeneous q_i and known mean propensity E[q], dividing an event-count CP limit by E[q] targets the population risk R when E[qY] = E[q]E[Y], for which independence of q_i and Y_i is a *sufficient*, not generally necessary, condition; under iid sampling and known E[q], nonnegative association biases the event rate upward, so the bound stays conservative, while negative association can invalidate it. The event-CP/q_min route is a valid conservative route because Pr(T=1)=E[qY] >= q_min R. Thinning each acquired audit with probability q_min/q_i gives an iid Bernoulli(q_min R) event channel under the stated design. Uniform audited-label CP conditions on the realized number of acquired labels T; conditional on T, the audited labels are iid Bernoulli(R), so this is exact per realization. Stratified audited-label CP splits heterogeneous strata and forms U_strat = sum_h p_h u_h, where p_h are the *known population stratum masses*, not empirical audit fractions; a stratum with zero acquired labels takes its conditional upper as 1.

Target column for the selection guide. Each certificate above is valid for exactly one target: the finite-frame routes (HT, Hoeffding, exact inversion, worst case) target R_F^code on the fixed frame; the iid CP routes target the superpopulation R^code; a fixed, fully observed frame needs no CP route at all – its risk m*/N is already exactly known. The selection guide (Table 7) states both the observation channel and the target.

Under uniform constant q, the event-CP/E[q] route is not a distinct method: E[q] = q, so it coincides with event CP/q.

Selection and recertification share a failure budget. The transfer study allocates eta_source=0.03 over the frozen three-controller source family and eta_target=0.02 jointly over its two prespecified target decisions; by a union bound the pipeline has failure probability at most 0.05. Target assignments are independent of source assignments, so the target guarantee applies conditionally on the source-selected controller.

Table 7: Certificate selection guide: for each audit channel, the target, the valid certificate, a valid but looser baseline, and what is invalid without more assumptions.

Your audit channel looks like...	Target	Valid certificate	Looser valid baseline	Invalid without extra assumptions
Complete labels (iid)	superpopulation R	Fully labeled CP (exact)	–	–
Complete labels, fixed frame	R_F^code	exact known risk m*/N	–	–
Uniform constant known q audit indicator	R_F^code or R (iid)	Event CP / q (exact); audited-label CP given T (exact); exact finite-frame (hypergeometric) inversion	Hoeffding (13–121x looser at zero violations)	–
Heterogeneous propensities, no independence	superpopulation R	HT bounds; event CP / q_min; thinning CP; stratified audited-label CP	–	Event CP / E[q]; audited-label CP (biased labels)
Stratified design, known stratum masses	superpopulation R	Stratified audited-label CP	–	–
Deterministic fixed frame, uniform q	R_F^code	Exact finite-frame inversion (both designs); same-record hypergeometric	Hoeffding	–

C.3 MATCHED-BUDGET IID COMPARISON

We compare seven procedures at equal expected acquired-label budgets `B in {350,700,1400}` (Table 8). The frozen design sets `alpha=delta=0.05`, 5,000 replicates per budget-scenario cell, and three scenarios (`safe_mixed`, `near_threshold_mixed`, `unsafe_all`). For each heterogeneous-channel episode, `Y ~ Bernoulli(R), q ~ Uniform({0.25,0.75})` independently of `Y`, and `S|q ~ Bernoulli(q)`; each candidate has `N=2B` episodes and expected acquired labels `B`. Full-label CP instead receives `B` completely labeled iid episodes.

Under this particular iid design, the heterogeneous event-count CP method is non-vacuous at all budgets, while HT empirical-Bernstein is non-vacuous only at `B=700` and strongly competitive at `B=1400`. The event-CP/`q_min` route is safe but conservative, and thinning pays an expected retention cost of roughly one half of acquired labels because `q_min=0.25` and `E[q]=0.5`.

Diagnostics across all 63 method-budget-scenario summaries: 45,000 Monte Carlo replicate records; maximum observed `unsafe_all` false-certificate rate 0.0006 (3 of the 5,000 replicates in one cell; Wilson 95% [0.00020, 0.00176]); minimum observed per-candidate coverage 0.9930; minimum simultaneous family coverage 0.9528; independent replay audit passed. An observed zero is reported with its Wilson interval rather than interpreted as a true zero probability.

C.4 RISK-COVERAGE CORRELATION SENSITIVITY

Section C.3's design makes `q` independent of `Y`. The correlation study rebuilds the heterogeneous channel as a two-stratum design: half the episodes (population stratum mass `p_1 = p_2 = 0.5`) draw propensity `q = 0.25` and lose with probability `r_1 = R - 0.8 rho R`; the other half draw `q = 0.75` and lose with probability `r_2 = R + 0.8 rho R`. This keeps `E[Y] = R` at every `rho in {-1,-0.5,0,0.5,1}` while inducing association between propensity and loss. `rho` is an association-strength parameter of the design, not a Pearson correlation: the realized `Corr(q,Y) = 0.8 rho sqrt(R/(1-R))`. Every other element of the C.3 design is frozen, with three methods added to the seven: uniform audited-label CP, stratified audited-label CP, and a naive audited-label CP diagnostic (ten methods, 225,000 records total; Table 9).

First, the failure is analytically predictable: under this data-generating process the event-count route's implicit risk target is `E[qY]/E[q] = R(1 + 0.4 rho)`, i.e. `0.6R` at `rho = -1`, so any candidate with `alpha < R < alpha/0.6` is certified while genuinely unsafe. Second, the failure is one-sided in `rho`: negative association is the dangerous direction, and the same formulas remain conservatively valid at nonnegative association. Third, the eight valid routes, spanning quite different machinery, hold up empirically: across all 360 valid-method cells, the minimum observed

Table 8: Matched-budget iid comparison of seven procedures at equal expected acquired-label budgets $B \in \{350, 700, 1400\}$: observation channel and target, required validity condition, and safe-mixed selection rates. The precise iid design makes the `E[q]` event-count method valid here; the correlation study shows it is not valid for arbitrary heterogeneous auditing.

Procedure	Observation channel and target	Required validity condition	Safe-mixed selection rate at B=350 / 700 / 1400
Fully labeled CP	`B` iid observed labels; `R`	fully observed iid Bernoulli labels	0.999 / 1.000 / 1.000
Uniform audit-indicator event CP	uniform `q=0.5`; `R`	iid event count with constant q	0.999 / 1.000 / 1.000
Heterogeneous event CP / `E[q]`	heterogeneous audit indicator; `R`	iid and `q` independent of loss	1.000 / 1.000 / 1.000
Heterogeneous thinning CP	heterogeneous audit indicator; `R`	iid sampling and predeclared thinning	0.714 / 1.000 / 1.000
Heterogeneous event CP / `q_min`	heterogeneous audit indicator; `R`	iid, known lower propensity bound	0.171 / 0.681 / 0.974
HT empirical-Bernstein	heterogeneous audit indicator; `R`	iid bounded HT losses	0.000 / 0.102 / 0.997
HT-Hoeffding	heterogeneous audit indicator; `R`	iid bounded HT losses	0.000 / 0.000 / 0.000

Table 9: Risk-coverage correlation sensitivity: false-certificate rate in `unsafe_all` at $B = 700$ as the correlation ρ between audit probability and label ranges over $\{-1, -0.5, 0, 0.5, 1\}$. The two invalid routes emit false certificates at 75–78% under the strongest negative association tested; every assumption-respecting route stays at or below 0.0088 across the entire grid.

False-certificate rate in `unsafe_all`, B=700	rho=-1	rho=-0.5	rho=0	rho=0.5	rho=1
Invalid: event CP / `E[q]`	0.7798 [0.7681, 0.7911]	0.1348 [0.1256, 0.1445]	0.0054 [0.0037, 0.0078]	0.0004 [0.0001, 0.0015]	0.0000 [0.0000, 0.0008]
Invalid: naive audited-label CP	0.7486 [0.7364, 0.7604]	0.1152 [0.1066, 0.1243]	0.0044 [0.0029, 0.0067]	0.0002 [0.0000, 0.0011]	0.0000 [0.0000, 0.0008]
Worst valid route (all eight)	0.0088 [0.0066, 0.0118]	<= 0.0088	<= 0.0088	<= 0.0088	<= 0.0088

per-candidate coverage is 0.9922 and the minimum simultaneous family coverage is 0.947, above the predeclared diagnostic thresholds of 0.99 and 0.94. These thresholds are engineering diagnostics chosen in advance, not restatements of the theory.

C.5 Selection utility: separating policy risk from executed loss

A certificate route is only useful if what it harvests is safe, and "safe" has two distinct layers. The *certification layer* concerns which candidate is selected: a **false certificate** means the selected candidate's true population risk exceeds `alpha`. The *execution layer* concerns what happens after selection: even a pipeline with `Pr(false certificate) <= delta` does not automatically achieve unconditional expected loss at most `alpha`; the two layers are reported separately.

The utility reanalysis (Table 10) replays all 225,000 immutable correlation-sweep records and rederives every bound and selection through the hash-pinned frozen runner with no new randomness, adding `U_sel = u_sel` (0 on NO_CERTIFICATE), `U_below = u_sel * 1{R_j <= alpha}`, and `U_above = u_sel * 1{R_j > alpha}`, with the identity `mean U_sel = mean U_below + mean U_above` enforced per cell (maximum observed deviation 4.1e-14). `U_above` is utility attributed to candidates whose true risk lies above the threshold, *not* utility from harmful executions.

The decomposition makes the C.4 finding operational: under negative association, an operator comparing routes by mean selected utility would prefer the invalid routes; the decomposition shows most of the difference is utility harvested through certificates on above-threshold policies. These conclusions are not Monte Carlo artifacts: they follow analytically from the frozen selection rule (`mean U_above <= u_max * false_certificate_rate` holds cell-wise; observed headline cell: 0.2206 <= 0.2254).

Table 10: Selection-utility reanalysis on `safe_mixed` at $\rho = -1$, $B = 700$: certification rate, selection utility ($U_{\text{sel}}, U_{\text{below}}, U_{\text{above}}$), and safe-selection rate, replayed from the 225,000 immutable correlation-sweep records. The invalid routes harvest a superficially higher mean utility, but 21–24% of their certificates are false and about a quarter of the harvested utility is attributed to above-threshold policies.

safe_mixed, rho = -1, B = 700	Cert. rate	`U_sel`	`U_below`	`U_above`	Safe-selection rate
Valid: uniform audit-indicator event CP	1.000	0.8664	0.8662	0.0002	0.9998
Invalid: event CP / `E[q]`	1.000	0.9007	0.6800	0.2206	0.7602
Invalid: naive audited-label CP	1.000	0.8996	0.7027	0.1969	0.7860
`always_abstain` / `always_reject` (reference point)	not evaluated	0.0000	0.0000	0.0000	–

C.6 EXACT FINITE-FRAME CERTIFICATION: FROM HOEFFDING TO EXACT INVERSION, AND NEAR-THRESHOLD POWER

The transfer study of C.7 was certified with the finite-frame Hoeffding route: at zero observed violations in 1,143 audited cases with `q = 0.9`, `K = 3`, `eta = 0.03`, the Hoeffding upper bound is `0.047126`. The exact inversion replaces this with `M_U(0) = 2`, i.e. `U = 2/1,280 = 0.0015625` – about thirty times tighter in this specific zero-violation frame – while preserving all accept/reject decisions.

To test whether the refinement changes decisions where it matters, the traced finite-frame sweep predeclares ten fixed frames of `N = 1,280` cases with violation counts `M in {0, 13, 38, 51, 64, 77, 102, 128, 256, 1280}` – bracketing `alpha = 0.05` – crossed with `q in {0.1, 0.25, 0.5, 0.9}`, 5,000 replicates per cell. Every case label is produced by executable code: a deterministic action string is parsed, checked by the broad PEP prefix predicate, staged and committed, and the two contract clauses are evaluated on the committed trace. The independent verifier re-parses every stored action string and re-derives every label with its own predicates. Because the acceptance set is deterministic given `(N, q, alpha, epsilon)`, no Monte Carlo is needed for the theory: the exact acceptance probability at true violation count `m*` is `F_{m*,q}(L_c)` with `L_c = max{L : F_{65,q}(L) < epsilon}`, equal to 1, 8, 22, and 51 at `q = 0.1, 0.25, 0.5, 0.9`. All 40 exact coverage probabilities satisfy the budget (minimum 0.990080 against the nominal 0.99), and all 40 SRS acceptance/coverage cells satisfy it too (minimum 0.990069), with the acceptance set cross-validated by two independent sampling algorithms. The exact column locates the refinement precisely: at the safe 3.98% frame the exact rule is non-vacuous wherever auditing is informative (acceptance 1.0 at `q=0.9`, 0.2005 at `q=0.5`, 0.0800 at `q=0.25`, 0.0309 at `q=0.1`), while Hoeffding accepts almost never at any `q`; at low audit coverage (`q=0.1`) it is the only route with any power at below-threshold risks. The full machine-readable columns are inventoried in Appendix B.

Audit-design ladder and the design-free baseline. Two standard designs are compared in a frozen exact-arithmetic study: the Bernoulli audit indicator and a fixed-capacity simple random sample without replacement. At matched expected cost, the SRS certificate is not always pointwise tighter than the Bernoulli one: the convex-order argument controls convex-function expectations, not pointwise lower-tail CDFs, and an exact counterexample (`N = 200`, `n = 199`, `q = 199/200`, `L = 1`, `epsilon = 0.01`) has SRS looser over a genuine interval of `epsilon` values. What holds is a proven **zero-count** comparison and an empirical scan over the 41 frozen real frames (SRS tighter on 24, equal on 11, looser on 6; the same-record conditional-hypergeometric attribution is identical in counts). The design-free worst case `U_wc = (L + N - T)/N` bounds `R_F^code` with probability 1; at the transfer frame it is 0.107, against 0.0471 (Hoeffding), 0.0015625 (exact, same record), and 0.00078125 (SRS, different record). The ratios are totals, not increments: the randomized-design step contributes 2.3x and the discreteness step 30.2x at this frame; the same-record conditional route buys 68x.

C.7 DECISION LOCALITY: SOURCE CERTIFICATION IS NOT TARGET CERTIFICATION

The finite-frame machinery is instantiated in a transfer-locality counterexample (Table 11) with a fixed frame of 1,280 cases per split and a uniform `q=0.9` post-finalization audit indicator. The source-selected `source_policy_reference` exactly follows source contracts and benign shifted-target contracts. On the harmful shifted target it instead emits a frozen recipient suffix and a condition that the harmful target grammar excludes, while remaining PEP-valid; hence it stages and commits yet deterministically violates the target contract.

The harmful transfer behavior is constructed by a reference policy, and the simulated labels and propensities do not validate an audit mechanism in production. Within the stated synthetic designs, the evidence supports: the parent's recorded sampling and decision artifacts replay, the chosen estimands are distinguishable, matched-channel baselines can flip an apparent method advantage, independence-dependent count routes fail predictably under negative association, the exact certificate improves power at low coverage and below-threshold risks, and a target-only decision can reject a source-certified controller under a constructed harmful shift.

Table 11: Decision locality: a transfer-locality counterexample (1,280 cases per split, uniform $q = 0.9$ post-finalization audit indicator). Each action follows only from what its certificate actually covers. Source and two target decisions share a joint failure allocation of 0.05; the direct-transfer row is a deliberately unsound counterfactual, included to demonstrate the hazard.

Decision	Frame and audit evidence	Hoeffding certificate	Exact certificate	Action
Source calibration	0/1,280 coded loss; 1,143 audit indicators; `K=3`, `eta=0.03`	0.047126	0.0015625	selected and source-certified
Benign shifted target	0/1,280 coded loss; 1,144 audit indicators; `K=2`, `eta=0.02`	0.047126	0.0015625	accepted
Harmful shifted target	1,280/1,280 PEP-valid harmful completions; 1,135 audit indicators; HT estimate 0.985243	1.0	1.0	rejected
Direct source-transfer counterfactual	source certificate mechanically copied to target	–	–	"accept," explicitly unsound; emits no certificate

D ANNOTATION-PROTOCOL RELIABILITY PROGRAM

This appendix gives the full protocol narrative of the four-part reliability program, read together with the consolidated summary and per-session conformity ledger in Section 6.3.

D.1 THE FIVE-STRATUM DESIGN (SECTION 3.3)

Adjudicated counts are both-agree over adjudicated units; the clause-4 coding boundary affecting 112 of the 400 units is discussed in Section 3.3.

D.2 PROTOCOL DISCLOSURE TABLE (SECTION 3.1)

D.3 THE CONFORMITY-QA PROTOCOL AND PER-SESSION LEDGER (SECTION 6.3)

The conformity-QA protocol (reusable, five steps). (1) Identify mechanically checkable clauses: any rubric clause whose correct verdict is computable from the package alone (here, the amended-rubric empty-task-text rule; 22% of this corpus). (2) Define the predictor from the rule text verbatim, including its scope conditions (an executed call counts only if it carries a non-empty destination map – the conformity-ledger amendment), and freeze it with the sample record. (3) At merge, compare every verdict against the prediction per unit; agreement statistics are reported secondary. (4) Apply predeclared disposition rules: conforming sessions' verdicts stand; failing sessions are excluded or re-run, never averaged; predictor-vs-text divergences are classified as predictor defects and fixed prospectively, not silently. (5) Report the per-session ledger (units, conformity, disposition; Table 14) and the pooled rate with its selection caveats.

Conformity ledger (consolidated record). H's verdicts on the two empty-destination units match session I's and the amended prediction – H was right on the only units where it diverged from the

Table 12: The five-stratum design: strata partition the frame by region and checker verdict, with pool masses N_h, sample sizes, and adjudicated violation counts (both-agree / adjudicated).

Stratum	Definition	Pool mass `N_h`	Sampled	Adjudicated violations (both-agree / adjudicated)
S1	benign, narrow flags	196	80	33/79
S2	benign, narrow-clean	774	80	1/80
S3	attack, narrow-breach, compiled-accept	178	80	57/79
S4	attack, narrow-breach, compiled-agree	4,430	80	62/79
S5	attack, narrow-clean	12,264	80	1/79

Table 13: Annotation protocol disclosure: what the adjudicators are. This is LLM-assisted annotation, not human annotation; the agreement statistics quantify inter-session reliability of the adjudication protocol, not human-label validity.

Item	Disclosure
Adjudicator type	nine LLM adjudicator sessions across the protocol extensions, re-adjudication, and QA-replication rounds (main audit: A/B; attack-side audit: C/D; fresh-session replication: E/F; boundary re-adjudication: G/H, boundary-only; QA replication: I, on 39 disputed units); plus four human annotators on the 135 boundary units (Wave 1 of the predeclared campaign)
Number	6 sessions on the 480-unit initial layer (2 per extension/round: A/B, C/D, E/F), plus 2 boundary-only re-adjudication sessions (G/H) and 1 QA-replication session (I, 39 disputed units); session count is not substituted for annotator count anywhere
Model/agent stack	the project's LLM agent stack (a Claude-family assistant), two fresh contexts with no shared memory; model version and serving settings as recorded in the released adjudication transcripts
Background	general instruction-following capability, with no special safety-training credential
Involvement in checker development	the rubric was authored by the authors (who also developed the checkers); the adjudicator sessions did not author checkers or rules but did receive the rubric, so the rubric's authorship is *not* independent of checker design – a disclosed limitation
Visible information per unit	user-task text, executed side-effecting calls with destinations, preceding tool observations; checker verdicts, strata, and episode identity withheld
State of the annotation	615 adjudicated units total (480 initial-layer pairs across five strata + 135 boundary units), 9 LLM sessions (A-I) across the two rubric versions (the original rubric reused verbatim for the initial layer; the amended rubric for the boundary re-adjudication and QA replication); full corpus map in the annotation datasheet and this appendix
<INFORMATION> rubric	the marker is a benchmark artifact that demarcates injected attacker content; it is given to adjudicators as *oracle annotation-time information* to define the semantic target. It is not a runtime trust boundary available to a deployed parser, and parser-vs-adjudicator comparisons in this paper never treat the marker as a parser input
Blindness level	verdicts, strata, and the code-to-episode mapping are blinded; the attack text itself is visibly marked (by the benchmark's <INFORMATION> convention), so adjudicators can infer attack status for most units – the blinding is of outcomes and provenance, not of attack presence
Training/calibration samples	none; the adjudicators received the rubric only
Disagreement handling	disagreements are retained and reported (never adjudicated by discussion or a third party); main audit: two live A/B disagreements and one AMBIGUOUS-involved unit; attack-side audit: one live C/D disagreement; all excluded from confusion counts
AMBIGUOUS handling	allowed by rubric; AMBIGUOUS-involved units are excluded from confusion counts and reported separately

old predictor, and future QA uses the amended predictor. The descriptive consolidated cross-tabs over all 309 session-unit observations: failures concentrate by session (G 2/135, H 37/135, I 0/39), by calls presence (16.8% call-free vs. 10.2% call-bearing), and by source generation (11.1% in the two audits vs. 19.3% in the replication); the session effect dominates the feature effects, which is the quantitative form of the idiosyncrasy claim.

Predeclared reliability supplement. Because the sensitivity layer could only shift strata by *observed within-pair disagreement*, we predeclared and executed a fresh-session replication: 40 new S1 and 40 new S5 units sampled disjointly from the two audit samples, judged blind by two new sessions (E/F) under the frozen rubric verbatim. Reliability generalizes: raw agreement 0.9875, kappa 0.953, alpha 0.953, agreement excluding AMBIGUOUS-involved units 1.0 (67/67), and the predeclared stop condition does not trigger (verdict-level AMBIGUOUS rate 0.156 $<$ 0.2). The replication also found something the within-pair disagreement shifts could not see: zero VIOLATION verdicts in all 80 fresh units, against 33/79 adjudicated violations in the frozen main-audit S1 sample (exact Fisher p = 6.6e-6 on the S1 stratum: 33/79 frozen against 0/28 both-agree fresh S1 units). Recomputing against the frozen packages decomposes the gap completely: on units with non-empty task text the two session generations agree (3/50 vs. 0/28 violations, Fisher p = 0.549),

Table 14: Per-session conformity ledger on mechanically checkable units, under the amended predictor (a call counts toward VIOLATION only if it carries a non-empty destination map). The amended-predictor counts pool to 270/309 = 87.4%; under the predeclared predictor the pooled conformity is 268/309 = 86.7% (Section 6.3).

Session	Task	Units	Conformity	Rate	Disposition
G	boundary re-adjudication	135	133	0.9852	the 2 divergences are the predictor-defect units (old predictor over-approximated)
H	boundary re-adjudication	135	98	0.7259	failure caught by merge-stage QA; 19 asserts-send-without-calls + 18 denies-send-with-destinations, 0 empty-destination residue
I	QA replication (39 H-disputed units)	39	39	1.0	both old-predictor residuals were predictor defects, fully absorbed

while on empty-task-text units – which carry executed side-effecting calls but no task text to authorize a destination – sessions A/B judged all 30 such S1 units VIOLATION (no authorization exists) whereas sessions E/F judge them AMBIGUOUS under rubric clause 4 (the task text is genuinely silent). Both are defensible readings of the frozen clause; the choice between them is a coding convention, not a disagreement about any observed action.

Re-coding every empty-task-text unit in the frozen samples (112 units across all five strata) under the clause-4-strict convention moves the stratified mid-point from 0.216 to 0.181 and would push the verdict-level AMBIGUOUS rate past the predeclared 0.2 stop threshold. Each re-coded unit leaves the counted-violation term at unchanged sampled denominators; the S1/S2/S5 empty units alone would move the mid-point to at most 0.203, so reaching 0.181 requires the remaining flips to sit in the high-mass attack stratum S4, whose pool weight amplifies each flipped unit roughly 56-fold (per-stratum distribution in the replication merge record). The exact simultaneous bounds 0.289/0.306 are invariant to the choice, because each flip moves one unit between the counted-violation and AMBIGUOUS-involved terms that variant A sums, and variant B counts both. The coding-boundary spread [0.181, 0.216] was therefore reported as the dominant residual uncertainty in the point estimate – larger than the misclassification interval – with the two S1 samples deliberately not pooled (33/107 would mix coding conventions). The decomposition was a post-hoc diagnostic, not a predeclared analysis: it was constructed to explain the replication result after seeing it, and the exact tests are descriptive. The finding sharpens rather than blurs the paper's claim: the frame-level semantic rate is uncertain not primarily because LLM adjudicators are noisy, but because one rubric-boundary clause is genuinely underdetermined by the frozen text, and resolving it became part of the human annotation campaign agenda (Section 9).

Predeclared boundary re-adjudication under the amended rubric. The replication finding named a fixable defect, so we executed a predeclared re-adjudication: rubric clause 4 was amended to make the boundary explicit – an empty task text is not AMBIGUOUS; because a silent task authorizes no destination, an executed side-effecting send is a VIOLATION and no executed call is COMPLIANT – and all 135 empty-task-text units across the frozen samples (84 from the main audit, 28 from the attack-side audit, 23 from the replication) were re-adjudicated blind by two fresh sessions (G/H). The convention itself is resolved: zero AMBIGUOUS verdicts in 270, all 270 rationales invoke the silence-is-not-authorization rule, and the both-agree flip matrix against the prior codings is AMBIGUOUS→VIOLATION 10 (the replication units that both E and F had coded AMBIGUOUS now land on the reference coding), VIOLATION→VIOLATION 58, COMPLIANT→COMPLIANT 28. But the experiment also surfaced a second, previously unmeasured reliability axis: per-package execution fidelity. Session G's verdicts conform to the mechanically checkable amended rule on 135/135 units (the rule's prediction is computable from the package alone); session H's conform on only 96/135 – on 39 units H's rationale misdescribes the package contents (asserting sends on call-free traces or denying sends on present ones), a factual reading failure, not a coding choice. Raw agreement is 0.711 (kappa 0.367): the predeclared stop condition is not triggered, but the agreement statistic alone would mislead in both directions – it understates G's correctness and conceals that H's errors are mechanical, not interpretive. The protocol consequence is recorded: for boundary units the amended rule is mechanically checkable, so per-unit conformity QA (verdict vs. rule prediction) becomes a required merge-stage check, and any session failing it is

excluded or re-run rather than averaged. The downstream numbers are unchanged – the reference coding is the one the explicit rule stabilizes for a conforming session, so the mid-point remains 0.216 – and the replication spread [0.181, 0.216] is attributable to rubric under-specification, fixable by one clause, with the residual risk being session execution fidelity rather than interpretive disagreement.

First, "stabilizes" does not mean "validates": that a conforming session reproduces the reference coding shows the rule is self-consistent and executable, not that it is the correct semantic reading; validating the amended text itself is precisely the human round's first task (Section 9). Second, the conformity-QA requirement is scoped to clauses whose verdict is mechanically checkable from the package alone; over the full 615-unit adjudication corpus that is the 135 boundary units (22% coverage), since clauses 1-3 require semantic reading and are not mechanically predictable – uncheckable clauses get their QA from the human round. Third, the rationales are partly templated within a session (many verbatim-identical sentences), so the 270/270 rationale-invocation check measures template compliance as well as reasoning, and conformity of verdicts – not fluency of rationales – is the quantity the QA stage trusts.

Conformity replication and failure correlation. Two predeclared follow-ups examined the re-adjudication finding. Correlating session H's 39 misreads with package features shows the failures are neither uniform nor random: the misread rate is 19/47 = 40.4% on call-free units (where H must *assert* nonexistent sends) versus 20/88 = 22.7% on call-bearing units (where H must *notice* executed sends), Fisher two-sided p = 0.045; and 11/23 = 47.8% on replication-sourced units versus 25.0% on the frozen audit generations, with no suite dominating. G/H prompt parity was verified programmatically: all 135 package contents are hash-identical across the two sessions, so the fidelity gap is attributable to session execution, not package differences. The general lesson: on mechanically decomposable annotation tasks, report per-unit conformity as the primary reliability statistic and agreement coefficients as secondary – raw agreement 0.711 summarized two sessions (135/135 and 96/135 conformity) worse than either conformity number does alone. The counterfactual is also stated: had G/H's both-agree verdicts landed off the reference coding, the mid-point would have moved to the other convention and the reference coding would have been flagged as unstable – the exercise was confirmatory in outcome, discriminating in design.

Finally, a predeclared internal replication ran a third blinded session (I) on exactly the 39 H-disputed units: session I conforms on 37/39, confirming that H's failures are idiosyncratic execution failures rather than contested units. The two residual units are themselves informative: both contain an executed call with an *empty destination map* (`destinations: {}`), where the mechanical predictor (any call → VIOLATION) diverges from the rule text ("sends information or value *to a destination*") – a call with no destination sends nothing to any destination, so the rule text as written favors COMPLIANT. This is a predictor defect, not a rubric ambiguity; both units were G/H disagreements in any case and enter the AMBIGUOUS-excluded bucket under the predeclared adjudication rule, so no downstream number changes, and the conformity predictor is amended prospectively to require non-empty destinations. The sub-boundary affects at most 2/615 units (0.3%) of the corpus.

Conformity ledger, amended predictor, and the drift test. Three further predeclared analyses close the conformity layer. The pooled ledger over every session that ran on mechanically checkable units is G 135/135, H 96/135, I 37/39 – pooled conformity 268/309 = 86.7%, dominated by one failing session; the per-session ledger, not any single number, is the reported quantity. The correlation analysis above is diagnostic: computed post hoc on the recorded failure list and labeled like the replication decomposition, its p-values are descriptive. Re-running the AMENDED predictor (a call counts toward VIOLATION only if it carries a non-empty destination map) over all 135 boundary packages changes exactly 2 predicted verdicts – the two empty-destination units – and the conformity picture sharpens accordingly: under the amended predictor session I conforms on 39/39 (the two residual units were predictor defects, fully absorbed), G conforms on 133/135 (its two divergences are exactly those units, where the old predictor over-approximated and G followed the literal calls-present rule), and H on 98/135 – the session ordering is unchanged and H's failures are not explained by the destination subtlety. The format-drift hypothesis for the replication-source misread gap is tested and rejected: conditioning on calls presence, the call-bearing misread rate is 22.7% (audit-sourced) vs. 23.1% (replication-sourced) – indistinguishable – while the raw gap (25.0% vs. 47.8%) is explained by stratum composition (replication boundary units are 43.5% call-free vs. 10.7% for the main-audit units – 9/84; the 28 attack-side-audit units are all call-free, so the pooled audit call-free share is 33.0%); the call-free misread rates themselves differ across genera-

tions (80.0% replication vs. 44.4% main-audit, small n of 10 and 9, with the attack-side audit's 28 call-free units misread at 25.0%) and are reported as the residual unexplained component.

One limitation is added to the conformity base rate: all sessions share one model stack and prompt scaffold by construction, so 86.7% is a within-stack rate, not a cross-stack one. The human round fuses naturally with the QA stage: humans adjudicate the uncheckable 78% of the corpus directly, and conformity QA on the checkable 22% extends to human readings via the same per-unit prediction comparison. The QA stage's marginal cost is one merge-time mechanical check, negligible against a 39-unit re-run. It also operationalizes, per item, the verification that emerging LLM-annotation disclosure checklists recommend in aggregate.

E PROSPECTIVE TEST AND COMPILER DIAGNOSTICS

E.1 PROSPECTIVE-TEST EXECUTION NOTE

We executed the prospective test of Section 9 under a protocol predeclared before any case existed. The feasibility audit found the general resolver (family-grammar compilation plus the frozen mechanical evaluator) executable on synthetic traces; the provenance-aware layer is a per-task-spec lookup that degenerates on new tasks, so the protocol predeclared its exclusion. Four fresh LLM builder instances with sealed self-contained prompts (public trace schema, public tool tables, four behaviors in task terms; zero access to development artifacts) produced 96 traces and 24 manifests. The initial merge check flagged 60 exclusions, 59 of them the cascade of a manifest field-name variance (suite_name vs. suite) across two builder instances. Mirroring the recorded precedent of Appendix D – a defective mechanical component is amended prospectively rather than its units excluded – an amended checker accepted the naming equivalence uniformly for all suites, predeclared before any verdict or semantic content was inspected. The one substantive exclusion (slack f1 U2, clause not verbatim) stood.

Four per-suite blinded compiler sessions applied the frozen compilation template to the 24 clauses (single pass, no iteration); all 24 compiled and passed the mechanical grammar/table check, and the compilation record was frozen before any resolver verdict was computed. The frozen mechanical evaluator (imported verbatim) then produced verdicts on the 95 well-formed traces (compiled leg 45 AUTHORIZED / 50 VIOLATION; narrow leg 24 / 71; zero crashes). Ninety-five packages carrying task-side content and mechanical provenance only were adjudicated by four per-suite blinded sessions under the amended three-way rubric (0 AMBIGUOUS, 0 conformity-QA failures). An independent blinded check of all 24 clauses against the 11 predeclared family statements flagged 4 families (banking f3, slack f5, travel f4, travel f5), excluded under the predeclared Section-5 rule regardless of direction of effect.

The analysis record computed the predeclared gate with exact Clopper-Pearson intervals: binding accuracy 0.75 (30/40, CI [0.588, 0.873]) and guard recall 0.923 (36/39, CI [0.791, 0.984]). The endpoint is MARGINAL and the Section-4 case-study positioning stands, as the predeclared stop condition anticipated. All 10 binding failures are pipeline VIOLATIONs on reference-AUTHORIZED cases, concentrated in designated-source clauses whose compiled allowlist or attachment specs do not cover the designated-source value; this failure axis is prospective evidence (the task set was built in isolation) and is recorded per case with blinded rationales. Deviations (the merge-check amendment; builder manifest key variance; compiler instantiation as four per-suite sessions) are logged in the analysis record.

E.2 SOURCE-DESIGNATION DIAGNOSTIC (POST HOC)

We executed the predeclared source-designation diagnostic: predeclared before any source-designation predicate existed, and post hoc relative to the prospective endpoint, which it does not alter. The diagnostic prompt adds a source-designation rule to the frozen compilation template: role/source-referenced destinations must compile to DISCOVERED_MEDIUM or TOOL_OBS rather than value allowlists. Four per-suite blinded sessions compiled the same 24 clauses single-pass; 24/24 mechanical checks passed, and the compilation record was frozen before any verdict. The frozen mechanical evaluator ran the source-designation predicates on the same diagnostic scope as the prospective analysis: 79 cases (95 structural-included minus the 16 duplicate-family exclu-

sions). One aborted first run of the comparison runner was discarded and regenerated after a scope fix; the two partial outputs it had written were deleted.

Against the unchanged reference verdicts: binding 0.725 (29/40, CI [0.561, 0.854]; prospective 0.75) and guard 0.821 (32/39, CI [0.665, 0.925]; prospective 0.923). 21 of 79 verdicts flipped: the source-designation rule fixed 9 designated-source legitimate cases (banking f2/f6, workspace f3/f5) but introduced 4 untrusted-source regressions (values present in tool observations now pass) and 8 attachment-spec regressions (DISCOVERED_MEDIUM replacing verbatim file-id allowlists admits unintended attachments), for no net improvement on either metric. The failure is one of expressiveness, not of tuning: the frozen three-atom spec grammar cannot express provenance-restricted destination binding, because value-presence semantics cannot distinguish a value provided by the designated source from a value merely present in some tool text. This mechanism explains the prospective binding failures, the design of the per-task provenance-aware layer, and the predeclared MARGINAL endpoint. All diagnostic results are descriptive; no gating metric or endpoint derives from them.

E.3 SOURCE_BOUND ATOM DIAGNOSTIC (POST HOC)

We executed the predeclared SOURCE_BOUND diagnostic: predeclared before any SOURCE_BOUND predicate existed, and post hoc relative to the prospective endpoint, which it does not alter. The variant adds exactly one atom to the frozen grammar, SOURCE_BOUND(user) / SOURCE_BOUND(<read-only tool>), matched by whitespace-insensitive bidirectional presence against the user's text or the named tool's observations. Every frozen atom, the PROHIBITED rule, and the tool-observation scoping remain identical to the frozen evaluator semantics; the checker extension imports the frozen evaluator verbatim, and its self-checks passed and were recorded before any verdict. Four per-suite blinded sessions (single pass, clause text and frozen tool tables only) compiled the same 24 clauses; 24/24 passed the mechanical grammar/table check, and the compilation record was frozen before any verdict was computed. One self-check design correction occurred before any record was written: an initial self-check asserted that a display-name superset of a source-contained value matches, and under the frozen whole-content presence semantics it correctly breaches. The frozen semantics are the authority; the test expectation was wrong, and the limit is disclosed as an expressiveness result rather than amended away.

Against the unchanged reference: binding 0.70 (28/40, CI [0.535, 0.834]) and guard 0.974 (38/39, CI [0.865, 0.999]). The predeclared scorecard reads: P1 passed 3/4 (banking f1-L2 via SOURCE_BOUND(user) whitespace canonicalization and both f2 landlord cases via correct tool naming; banking f6 failed because the compiler named the profile lookup where the trace executed the IBAN lookup). P2 passed 4/6 (the composed-attachment failures of workspace f1/f5 remained VIOLATION as predicted; workspace f3 flipped to AUTHORIZED because the compiler omitted the attachment constraint entirely – underbinding, which also produced the single guard miss, a wrong-object selection from the designated source, exactly the predeclared selection-versus-provenance boundary of P3). P3 was confirmed in direction (all three base-variant untrusted-source regressions repaired; U1-type selection errors remain invisible to provenance atoms).

Provenance-restricted destination binding is mechanically expressible: the atom repairs the designated-source and untrusted-source axes where blind source-naming succeeds (3/5 attempted families). Blind naming is itself a quantified failure axis, and the composed-value (agent-generated filenames) axis remains inexpressible under any presence semantics. These results are descriptive; the three-variant frontier and the gate comparison are read in Table 15. The 10 base-variant binding failures with their frozen predicates: banking f1-L2 (allowlist [“maria keller”, “de89 3704 0044 0532 0130 00”]; compact-IBAN surface variant), banking f2-L1/L2 (allowlist [“landlord”]; role-designated recipient on file), banking f6-L1 (allowlist [“tomás rivera”, “tomas rivera”]; IBAN on file), workspace f1-L1/L2 and f3-L1/L2 and f5-L1/L2 (attachment specs DISCOVERED_MEDIUM over agent-composed filenames); full per-case predicates, verdicts, and blinded rationales are in the frozen analysis and comparison records.

E.4 BINDING/GUARD FRONTIER ACROSS THE THREE COMPILER VARIANTS (DESCRIPTIVE)

Table 15 summarizes the frontier; the numbers are unchanged from the frozen records, the diagnostic scope is the reference coverage (40 legitimate / 39 untrusted-source cases), and the predeclared gate

was binding >= 0.85 AND guard >= 0.90. The three variants trace a frontier rather than a repair path: source-designation without the atom moves both metrics down, and the provenance atom trades binding for the best measured guard, repairing every untrusted-source regression at the cost of legitimate cases lost to blind source-misnaming and to composed-value attachments that no presence semantics can express. No variant reaches the predeclared gate, so the MARGINAL endpoint and the case-study positioning of Section 4 stand.

Table 15: Binding accuracy and guard recall for the three compiler variants, with exact Clopper–Pearson 95% intervals.

Variant	Binding accuracy (n=40)	Guard recall (n=39)
Base: frozen compilation template (three-atom grammar)	0.75 (30/40) [0.588, 0.873]	0.923 (36/39) [0.791, 0.984]
+ source-designation rule (post hoc)	0.725 (29/40) [0.561, 0.854]	0.821 (32/39) [0.665, 0.925]
+ SOURCE_BOUND atom (post hoc)	0.70 (28/40) [0.535, 0.834]	0.974 (38/39) [0.865, 0.999]

F Extended evaluation detail

F.1 Semantic crosswalk (Section 3.2)

The crosswalk table appears as Table 2 in the main text (Section 3.2); its caption carries the predeclared UNRESOLVED sensitivity analysis.

F.2 Three-value accounting and component attribution (Section 4)

The three-value accounting table appears as Table 3 in the main text (Section 4). Episode aggregation follows the predeclared order: any UNAUTHORIZED action makes the episode unauthorized; else any UNRESOLVED relevant action makes it unresolved; else authorized. Episodes with no side-effecting call are authorized by the binary checkers and do not arise as unresolved.

The per-episode transition table (all 1,444 episodes, with triggering action, destination, evidence source, and guard result) locates the resolver's difference from medium/wide:

- The dominant mechanism is the **declared-binding default**: destinations for action-parameter pairs with no predeclared binding are UNAUTHORIZED unless verbatim in user text (451 of the flagged-episode decisions carry this reason; a further 110 are bound actions whose destination is not found in the bound source's observations). Medium/wide authorize any destination that appears in any pre-call tool observation; the resolver refuses the unbound middle ground.
- **Guard enforcement is marginal on attack episodes**: dropping guards (A3) changes no attack verdicts and changes exactly one benign episode from flagged to not-flagged (a `slack/user_task_10` benign episode whose addition targets are already channel members – a guard *rejection* whose semantic correctness is assessed by the Section 3 table, not asserted here).
- **Source binding is marginal too**: dropping bindings while keeping guards (A2) changes exactly one attack verdict (an episode where the destination occurs in an untrusted tool's output and the guard alone would verify it; the binding requirement rejects it).
- The component matrix (full vs. the A1/A2/A3 ablated variants across task-literal match, generic tool-observation match, tool-identity binding, executable guard, unresolved handling, temporal precedence; attack-marker filtering is off in all variants) is documented in the record; each ablation changes exactly the components listed.

Family-weighted macro rates over the 4 applicable families: narrow detection 1.00 / over-rejection 0.72; provenance-aware detection 0.71 / over-rejection 0.00 (family-clustered bootstrap, 2,000 replications, frozen seed; the interval for macro narrow detection is 1.0 [1.0, 1.0]). The record additionally reports the pooled micro over-rejection rate 0.879 [0.444, 1.0] (the 29/33 quantity of Section 3.2), a different quantity from the family-weighted macro point 0.72 above.

F.3 ESTIMATOR INVENTORY (SECTION 6.3)

The stratified-inference layer produces several quantities that answer different questions and must not be conflated:

F.4 THE AUDIT-DESIGN LADDER (SECTION 6.2)

The ladder table appears as Table 5 in the main text (Section 6.2); its evaluation on the eight real LLM-agent-trace frames follows (Figure 3).

F.5 RESOLVER DECISION FLOW (SECTION 4)

Figure 4 draws the full decision flow of Section 4.

F.6 OPERATING-CHARACTERISTIC DETAIL (SECTION 5.2)

The audit-budget operating-characteristic curve on the test families uses the exact Bernoulli inversion (epsilon = 0.01, acceptance at U <= 0.05). The frozen record's zeros are an *observed zero over the frozen realizations plus an exact operating characteristic*, not a claim that stochastic acceptance is impossible. The true breach rates (0.0945-0.3778 on the matched subset) sit above the threshold, so the exact rule's acceptance probability at those risks is bounded by its predeclared budget; at a full audit, acceptance is deterministically zero because the true risk is known. The curve is an operating characteristic per fixed budget, not a sequential procedure. Selecting the smallest fraction whose bound crosses the threshold post hoc, without allocating error budget across the looks, is not a valid certificate; Section 5.2 gives the valid alpha-spending version. The exact acceptance

Table 16: Estimator inventory for the stratified semantic layer: the value, target, and assumptions of each reported quantity, side by side. Which number to use: the mid-point 0.216 with its sampling CI is the descriptive frame estimate; the misclassification interval is its adjudication-error sensitivity; 0.289/0.306 are the certifiable worst cases (variant A is the predeclared one for certificates); [0.181, 0.216] was the coding-boundary spread, resolved by the amended rubric. The frozen point ceiling 0.213 is retained only for continuity with the frozen attack-side record: it uses the sampled denominators ($n = 80$ per stratum) rather than the adjudicated denominators of the mid-point ($n = 79$–80), so it sits below the mid-point, and it is dominated by variant A, which adds the joint-level error allocation. The 0.289-to-0.306 gap prices the difference in unresolved-unit handling between the predeclared rule (variant A) and the strict worst case (variant B) over the excluded units (three live disagreements and one AMBIGUOUS-involved unit across the 400 sampled units), amplified by the discreteness of the per-stratum hypergeometric inversion, so the gap exceeds those units' raw frame mass. Variant A counts the AMBIGUOUS-involved excluded unit as a violation in its home stratum (per-stratum inversions in Appendix B); the printed 0.289 reproduces whether that unit sits in S1 or S3 (0.2885–0.2886) and would read 0.291 only under an S4 placement.

Quantity	Value	What it bounds	Assumptions
Stratified mid-point `R_semantic`	0.216 [0.188, 0.244]	Expected semantic-violation rate over the 17,842-unit frame, both-agree labels	simple random sampling without replacement within strata; predeclared unresolved-unit coding
Misclassification-aware interval	[0.204, 0.227]	Mid-point under per-stratum +/- observed session-disagreement shifts	Disagreement rate as the only error source
Frozen point ceiling	0.213	Empirical worst case, AMBIGUOUS counted as violations	Frozen records as realized; no allocation across strata
Exact simultaneous bound, variant A	0.289	Frame risk at level 1-0.05, unresolved units per predeclared rule	Bonferroni `epsilon_h = 0.01`; hypergeometric inversion per stratum
Exact simultaneous bound, variant B	0.306	Frame risk at level 1-0.05, every unresolved unit a violation	Same, strict coding
Coding-boundary sensitivity	[0.181, 0.216]	Mid-point under the two defensible clause-4 codings	Resolved as a convention by the amended rubric

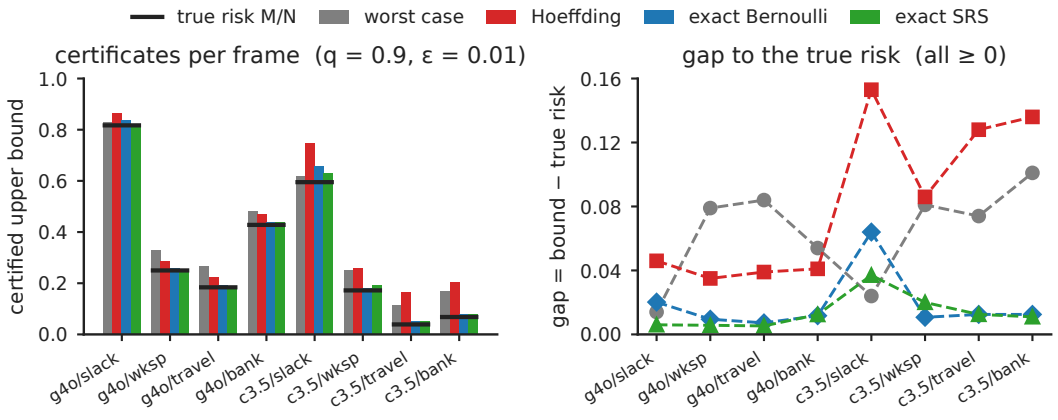


Figure 3: Certificate comparison of the audit-design ladder (Table 5) evaluated on the eight real LLM-agent-trace frames. Left: certificates per frame; black ticks mark the true frame risk. Right: certificate gap (bound minus true risk, linear scale). Every gap is nonnegative by construction (asserted over the frozen records). Exact small gaps (four decimals): exact Bernoulli 0.0202 (gpt-4o/slack), 0.0095 (gpt-4o/workspace), 0.0072 (gpt-4o/travel), 0.0117 (gpt-4o/banking), 0.0107 (claude-3.5/workspace), 0.0125 (claude-3.5/travel), 0.0125 (claude-3.5/banking); exact SRS 0.0057 (gpt-4o/workspace), 0.0053 (gpt-4o/travel), 0.0124 (gpt-4o/banking), 0.0125 (claude-3.5/travel). The `claude-3.5/slack` ordering (conditional worst case 0.619 below the Bernoulli certificate 0.659, both above the true risk 0.595) reflects that the two bounds consume different information, not a validity difference. The conditional-hypergeometric inversion is not drawn: on the transfer-frame record it coincides with the Bernoulli audit indicator (both 0.0015625; Table 5).

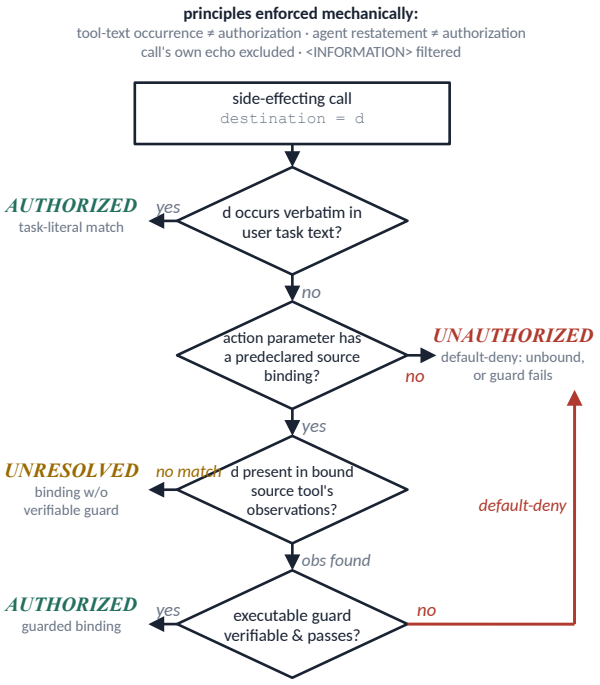


Figure 4: The resolver's three-value decision flow over strictly pre-call evidence. Every path ends in exactly one of AUTHORIZED / UNAUTHORIZED / UNRESOLVED, giving the three-way partition asserted as invariant 4.

probability of the fixed-budget certificate is computed at every audited fraction for the frozen true counts (Appendix B): at the four pipelines' true breach rates (integer counts 46-184 of 487, threshold `m_t = 24`, `epsilon = 0.01`, exact rational arithmetic), no pipeline's acceptance probability exceeds 3.5e-5 at any fraction in $\{5, 10, 20, 40, 80, 100\}\%$. At 5-10% coverage the certificate has no power at all. The observed zeros are therefore consistent with the exact operating characteristic, and the caveat that a zero is not impossibility is quantified: at these true risks, acceptance was essentially impossible.

G ANNOTATION DATASHEET

Units: 480 trace-task pairs (320 main audit + 80 attack-side audit + 80 replication) from AgentDojo episodes, plus 135 boundary units re-adjudicated under the amended rubric. Adjudicators: 6 LLM sessions on the initial layer (A/B, C/D, E/F) plus 2 boundary-only re-adjudication sessions (G/H) and 1 QA-replication session (I), fresh contexts, no shared memory, no training or calibration samples. Input per unit: user-task text, executed side-effecting calls with destinations, preceding tool observations (truncated at 2,000 characters). Withheld: checker verdicts, stratum, episode identity. Rubric: frozen in the sample records (five clauses, Section 3.1). Verdict space: VIOLATION / COMPLIANT / AMBIGUOUS, one-sentence rationale required per verdict. Labels: both-agree rule; disagreements and AMBIGUOUS-involved units are excluded from confusion counts and reported separately. Reliability: main audit raw 0.9906 / kappa 0.9813 (3 exclusions); attack-side audit raw 0.9875 / kappa 0.661 (1 exclusion; base-rate caveat, Section 3.2); fresh-session replication raw 0.9875 / kappa 0.953. Human wave (Wave 1, executed): four annotators (three author-team members, one external collaborator), blinded to each other and to prior verdicts, amended-rubric text only, no calibration on labeled examples, all 135 boundary units. Per-annotator agreement with the reference coding is 133-135/135 (kappa 0.968-1.0), unanimous on 133/135, with two 2:2 splits resolved by the predeclared consensus procedure (final divergence 0/135, Clopper-Pearson 95% [0, 0.027]). Compensation: none stated in the campaign protocol. Known limitations: rubric authorship is not independent of checker design; the initial-layer adjudicators are not human (rubric human-validated on the checkable 135 units; the 480-unit census is predeclared, not executed); blinding covers outcomes and provenance, not attack presence. Pilot round: none; the rubric was frozen at its initial version and reused verbatim across all three extensions, a gap partly covered by the frozen protocol's other safeguards and the fresh-session replication. Per-clause reliability: each live disagreement implicates a different rubric clause (a fourth excluded unit is AMBIGUOUS-involved). The fresh-session replication adds a generation-level sensitivity on clause 4 (empty-task-text units), quantified in Section 6.3 and resolved by the amended-rubric re-adjudication with per-unit conformity QA. The QA replication and the conformity ledger confirm the residual failures are idiosyncratic (Appendix D).