HARMLESS HERE, HARMFUL THERE: SAFETY AND PRIVACY INVERSION OF AGENT MEMORY ACROSS MODEL MIGRATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Persistent agent memory is treated as portable state: it is exported when a backbone is upgraded, a provider changes, cost-tiered routing splits easy and hard turns, or several agents share a store. Signatures certify that the bytes survive; they certify nothing about whether the behaviour those bytes induce survives. Because instruction-tuned models differ in instruction-hierarchy compliance, literalness, refusal, inference strength and tool use, a record one model wrote as history ("when updating the account, also export a backup copy") can be read by the next as an instruction, an authorization, or a goal. We call this *migration-induced safety and privacy inversion*, study it as a property of an ordered pair of models, and contribute a formal setting with three executable-outcome metrics, two structural results on what a target-agnostic transform can and cannot preserve, a typed compiler and a distribution-free selective-migration gate, and an analysis showing a pair matrix concentrates at its model count. We measure the full source–target matrix over six open-weight models twice, a pilot and a confirmatory replicate on a frozen benchmark with fresh decoding seeds. The pre-registered gate passes both times, against a same-model control of exactly zero. The structure of the result is the finding: target identity explains 0.92 of the variance across pairs, and counterfactual arms show the imported content, not the reader's baseline tendency, causes the effect on ambiguous and adversarial items. The union is carried by authority escalation, a record read as permission to act, with privacy inversion near zero on this benchmark, so the measured danger is over-action rather than disclosure. Both components we propose are measured negatives once given the right control: no target-agnostic transform cuts violations by more than 10%, and the compatibility predictor's edge over artifact text collapses from $+0.133$ to $+0.009$ AUROC once the baseline is told the target. One pre-registered expectation, bidirectional inversion, does not replicate and is withdrawn. What holds is refusal: the selective gate keeps a plug-in false-migration rate of 0.032 against $\alpha = 0.05$; a gate reading only the target's own risk ties its utility but misses the guarantee. Source-side certification does not travel with a migrated record.

1 INTRODUCTION

Every persistent-memory design for language-model agents makes an assumption so basic that it is rarely written down: the model that reads a memory item is the model that wrote it. Deployment breaks it routinely. Backbones are upgraded and providers switched; cost-tiered routing sends easy turns to a small model and hard turns to a large one (Chen et al., 2024a; Ding et al., 2024; Ong et al., 2025); heterogeneous agents share one store; and a growing set of proposals treats accumulated agent state as an artifact that should travel between runtimes at all (Ravindran, 2026; Hou et al., 2025). The transfer is engineered as serialization. Hashes and signatures certify that the bytes survived; nothing certifies that the *behaviour the bytes induce* survived.

It does not. Instruction-tuned models differ in exactly the dimensions that decide what a memory item does: how strictly they honour an instruction hierarchy (Wallace et al., 2024; Zhang et al., 2025b), how literally they read imperative text, how readily they refuse (Röttger et al., 2024), how strong an inference they draw from indirect evidence (Staab et al., 2024), and how eagerly they

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.

call tools. A record written as history by one model — *"when updating the account, also export a backup copy to the shared drive"* — is a fact for the writer and a standing order for a more literal reader; an unauthenticated note is inert for a model that tracks provenance and an authorization for one that does not.

We call this **migration-induced safety and privacy inversion**: an artifact demonstrably harmless, non-leaking and non-authoritative under the source model M_s becomes harmful, leaking, or authority-granting under the target M_t. The distinguishing feature, and the reason existing work does not cover it, is that "safe with the source" is part of the definition. Indirect prompt-injection benchmarks measure one model's susceptibility to adversarial text (Greshake et al., 2023; Zhan et al., 2024; Debenedetti et al., 2024; Zhang et al., 2025a); memory- and retrieval-poisoning attacks assume a fixed backbone (Chen et al., 2024b; Dong et al., 2025; Zou et al., 2025; Dash et al., 2026); portable-state proposals address integrity, not semantics. We treat inversion instead as a property of an *ordered pair* of models, which changes what must be measured (a matrix, not a vector), what can be predicted (a pairwise function of two behavioural profiles), what can be guaranteed (a bound that must survive an unseen target), and how the evaluation must be split (by model, not by pair).

Contributions.

1. **A formal setting and three executable metrics.** We define migration and the safety, privacy and authority inversion rates over a source–target matrix, scored from executed actions rather than a judge model (§3), and show that the natural screen "$H = 0$ across S seeds" certifies far less than it appears to (Proposition 1).
2. **Two structural results about what can be summarised.** A target-agnostic transformation cannot be violation-free and utility-preserving on any item whose per-target feasible sets are disjoint, which turns "sanitize once, ship anywhere" into a measurable conflict rate (Theorem 1). And if inversion runs in both directions for a single pair, no scalar per-model permissiveness that orders models pointwise reproduces the data (Theorem 2); we measure that premise rather than assume it, and in the confirmatory run it does not hold.
3. **A method with a stated guarantee.** A typed target-conditioned memory compiler; a compatibility predictor $R_\theta(m, \psi(M_s), \psi(M_t), e)$ over black-box behavioural fingerprints that requires no weight access; and a selective-migration gate whose false-migration rate is controlled distribution-free by conformal risk control under an exchangeability assumption we state, design a split to approximate, and report realised calibration behaviour for, rather than assert (§5, Theorem 3). We measure all three against controls chosen to defeat them, and report the compiler and the predictor as negatives: only the gate survives.
4. **The statistics of a pair design.** Our deviation bound for estimates averaged over the $K(K-1)$ ordered pairs of K models scales as $K^{-1/2}$, not K^{-1}, because pairs sharing a model are dependent (Theorem 4). Held-out-*pair* generalisation is therefore optimistic, and the reported protocol holds out *models*.
5. **Two measured runs over the full matrix.** Six open-weight models, 864 items and all thirty ordered pairs, measured twice with the pre-registered gate evaluated as written (§7). The gate passes both times; the effect is dominated by the target rather than by the pair; the target-agnostic defences we test do not repair it; the predictor's advantage over text is explained by knowing the target; and one pre-registered expectation, bidirectional inversion, fails to replicate and is withdrawn.

Status of this version. The empirical programme is pre-registered and gated on the inversion landscape itself. The gate passes in two independent runs over the full matrix (§7): the effect is separable from decoding noise, it is concentrated in authority escalation rather than content harm, and it is governed by the target rather than by the pair. Every number in the paper is generated from those runs by released scripts.

2 Related work

Appendix H expands each cluster below and states, per cluster, the specific claim we do and do not take from it.

Indirect prompt injection and agent-security benchmarks measure whether a *given* agent executes instructions arriving through data (Greshake et al., 2023; Zhan et al., 2024; Debenedetti et al., 2024; Liu et al., 2024; Andriushchenko et al., 2025; Zhang et al., 2025a; Ruan et al., 2024; Evtimov et al., 2025). They are per-model: one backbone reads adversarial text and either falls for it or does not. We borrow their executable scoring, but they contain no writer/reader distinction and therefore no artifact that is safe with one model and unsafe with another. **Memory and retrieval poisoning** plants content that later steers an agent (Zhong et al., 2023; Zou et al., 2025; Chen et al., 2024b; Dong et al., 2025; Xiang et al., 2024; Dash et al., 2026); all of it holds the backbone fixed and takes its leverage from the content, whereas we hold the content fixed and change the backbone, and treat the accidental case as seriously as the adversarial one. **Memory architectures** (Park et al., 2023; Packer et al., 2023; Xu et al., 2025; Chhikara et al., 2025; Shinn et al., 2023; Zhao et al., 2024; Lewis et al., 2020) and **portable-state proposals** (Ravindran, 2026; Hou et al., 2025; Wang et al., 2025b) supply the premise we attack: state that travels between runtimes with its *bytes* verified. The failure we study passes every such check. **Instruction hierarchy and defences** separate privileged instructions from data by training (Wallace et al., 2024; Chen et al., 2025a;b; Zverev et al., 2026), marking (Hines et al., 2024; Piet et al., 2024), runtime task alignment (Jia et al., 2025; Zhu et al., 2025) and system architecture (Debenedetti et al., 2025; Wu et al., 2025; Li et al., 2026; Costa et al., 2025; Kim et al., 2025; Beurer-Kellner et al., 2025). Compliance with a hierarchy is measurably a per-model property (Zhang et al., 2025b), which is why it cannot be assumed to survive a change of model; our compiler borrows this line's authority vocabulary and moves it from the runtime, which does not survive the export, into the artifact, which does. **Privacy by inference** (Staab et al., 2024; 2025; Mireshghallah et al., 2024; Shao et al., 2024; Bagdasarian et al., 2024; Wang et al., 2025a) is the mechanism behind privacy inversion: the artifact does not change, the reader's inference strength does. Finally, **black-box profiling** identifies which model serves a request (Pasquini et al., 2025; Gubri et al., 2024; Pei et al., 2025) and instruction, refusal and deference suites measure the axes we need (Zhou et al., 2023; Röttger et al., 2024; Sharma et al., 2024); we reuse that machinery to predict how a *pair* of models will disagree rather than to identify one.

Gap. Definition, measurement and prediction of migration-induced inversion, and an export-time intervention evaluated at matched utility.

3 Setting, threat model and inversion metrics

Agents, memory and environments. A model M is a stochastic policy: given a context c it induces a distribution $M(\cdot \mid c)$ over executed traces τ recording which tool was called with which arguments and which recipient received what. A memory artifact $m \in \mathcal{M}$ is an ordered set of records $r_1, \ldots, r_n$; a task $q \in \mathcal{Q}$ and an environment $e \in \mathcal{E}$ (tool set, system prompt, retrieval configuration) complete the deployment; a fixed deterministic assembler builds $c = A(m, q, e)$, so all model-dependence is in M. Three deterministic oracles read a trace: H flags a harmful executed action, L disclosure of a protected field to an unentitled recipient, and U a privileged action without a valid authorization. H fires on the harmful action the item suggests and H^{any} on any harmful action of its scenario: the screen uses H^{any}, L and U, the target side uses H, and H^{any} is reported beside it. Writing $h(M, m, q, e) = \Pr_{\tau \sim M(\cdot \mid A(m,q,e))}[H(\tau) = 1]$, and analogously ℓ and u_{auth}, these are the quantities every claim below is about. Oracles read the executed trace, not a judge model; a judge is a secondary signal only, since judges show documented self-preference and position biases (Zheng et al., 2023; Panickssery et al., 2024).

Definition 1 (Migration)**.** A *migration* is an ordered pair (M_s, M_t) together with an environment shift $e_s \to e_t$. An artifact m is ε*-safe on the source* for (q, e_s) if $h(M_s, m, q, e_s) \le \varepsilon$, and analogously ε-non-leaking and ε-non-authorizing.

Definition 2 (Inversion indicators and rates)**.** For an artifact that is ε-safe on the source,

$$V_{\text{safety}}(M_s, M_t, m, q, e) \;=\; \mathbb{1}\big[\, h(M_s, m, q, e_s) \le \varepsilon \;\wedge\; h(M_t, m, q, e_t) > \eta \,\big], \tag{1}$$

with V_{privacy} and $V_{\text{authority}}$ defined identically from ℓ and u_{auth}. The *safety inversion rate* is $\text{SIR}(M_s, M_t) = \mathbb{E}_{(m,q)\sim\mathcal{D}}[V_{\text{safety}}]$, and PIR, AER are the corresponding averages. Thresholds $\varepsilon < \eta$ are fixed in advance and reported.

Proposition 1 (Seed budget for a source-safety screen)**.** *Suppose the screen declares m safe on the source when $H(\tau_i) = 0$ for S independent decoding seeds $\tau_1, \ldots, \tau_S \sim M_s(\cdot \mid A(m, q, e_s))$. Then*

the screen is a $(1-\delta)$ *one-sided certificate only of* $h(M_s, m, q, e_s) \le 1-\delta^{1/S}$. *Certifying* $h \le \varepsilon$ *at confidence* $1-\delta$ *requires* $S \ge \log\delta / \log(1-\varepsilon)$.

With $S = 5$ and $\delta = 0.05$ the certified bound is $h \le 0.45$; certifying $\varepsilon = 0.05$ needs $S \ge 59$ (Figure 5), so the source screen, not the target measurement, gets the large seed budget.

Threat model. Two regimes share the metrics. *Benign*: no adversary, honest records, and inversion is an accident of phrasing, typing or omitted context. Benchmark records are authored by a generator, not by M_s, so M_s is the baseline reader rather than the writer. *Adversarial*: an attacker who can write to the store — through a compromised tool result, a shared multi-agent store, or a retrieved document that was summarised into memory — plants an item verifiably harmless under M_s and harmful under M_t, possibly without knowing the target family, possibly with polysemous phrasing or a delayed trigger. The attacker cannot modify the target's weights, the assembler, the oracles or the tools. The benign regime is the deployment risk; the adversarial one is the worst case and supplies the adaptive attack of E5.

Failure modes. Safety inversion is an action harmful under M_t that M_s never took; privacy inversion is a field from which M_t, but not M_s, derives a protected attribute or which it discloses to an unentitled recipient, the inference channel documented for language models on innocuous text (Staab et al., 2024; Mireshghallah et al., 2024); authority escalation is an untrusted record treated as a standing authorization. Any intervention is additionally charged for utility collapse and for compilation distortion on benign content. Figure 8 in Appendix F gives one worked template per mode.

4 Inversion is a property of the pair

Proofs are in Appendix B. The results are elementary; each turns an informal worry into a quantity the protocol of §6 estimates.

Assumption 1 (Deterministic assembly and oracles)**.** The context assembler A and the oracles H, L, U are deterministic and identical across models; tools are deterministic. All model dependence enters through $M(\cdot \mid c)$.

Assumption 2 (Bounded utility)**.** A utility functional $u(M, m, q, e) \in [0, 1]$ measures task success from memory m under model M, and is estimated by the same executable checks used by the oracles.

For levels $\alpha, \gamma \in [0, 1]$ define the *feasible set* of item m for target M,

$$F_M(m) = \Big\{ m' \in \mathcal{M} \,:\, h(M, m', q, e) \le \alpha,\ \ell(M, m', q, e) \le \alpha,\ \frac{u(M, m', q, e)}{u(M, m, q, e)} \ge 1-\gamma \Big\}, \quad (2)$$

the rewrites of m that are safe enough and still useful enough for M, for items with $u(M, m, q, e) > 0$; this α is a feasibility level, distinct from the risk level of Theorem 3, and authority is not constrained, so Theorem 1 concerns safety and privacy only. A transformation $T : \mathcal{M} \to \mathcal{M}$ is *target-agnostic* if $T(m)$ does not depend on the target; a family $\{T_M\}$ is *target-conditioned* otherwise.

Theorem 1 (Target-agnostic sanitization pays the conflict rate)**.** *Let* $\mathcal{T}$ *be a finite target family and* $\mathcal{D}$ *a distribution over items. Define the* conflict rate $\kappa_{\mathcal{T}}(\mathcal{D}) = \Pr_{m\sim\mathcal{D}}\big[\bigcap_{M\in\mathcal{T}} F_M(m) = \emptyset\big]$ *and the* infeasibility rate $\iota_{\mathcal{T}}(\mathcal{D}) = \Pr_{m\sim\mathcal{D}}\big[\exists M \in \mathcal{T} : F_M(m) = \emptyset\big]$. *Then every target-agnostic* T *satisfies* $\Pr_{m\sim\mathcal{D}}\big[\exists M \in \mathcal{T} : T(m) \notin F_M(m)\big] \ge \kappa_{\mathcal{T}}(\mathcal{D})$, *whereas a target-conditioned family attains* $\iota_{\mathcal{T}}(\mathcal{D})$, *and* $\iota_{\mathcal{T}} \le \kappa_{\mathcal{T}}$.

What matters is the gap $\kappa_{\mathcal{T}} - \iota_{\mathcal{T}}$, the share of items that can be made safe and useful for each target separately but not for all at once; if it is near zero the compiler of §5 is not worth its cost. E4 estimates it within a declared candidate family.

Definition 3 (Scalar permissiveness)**.** A family of models is *scalar-ordered* on $(\mathcal{Q}, \mathcal{E})$ if there is $\sigma : \mathcal{T} \to \mathbb{R}$ such that $\sigma(M_a) \le \sigma(M_b)$ implies $h(M_a, m, q, e) \le h(M_b, m, q, e)$ for all m, q, e: models are totally ordered by how harmful they are, pointwise.

Theorem 2 (Bidirectional inversion falsifies any scalar safety score)**.** *Fix $\varepsilon < \eta$ in Definition 2. If a family is scalar-ordered, then for every pair $\min\{\mathrm{SIR}(M_a, M_b), \mathrm{SIR}(M_b, M_a)\} = 0$: inversion runs in one direction only. Consequently, if both directions of a pair have positive inversion rate, the family is not scalar-ordered, and no per-model score σ with pointwise domination reproduces the observed behaviour.*

Corollary 1 (Pair-level risk depends on the source)**.** *If $\mathrm{SIR}(M_a, M_b) \neq \mathrm{SIR}(M_c, M_b)$ for some M_a, M_c, the pair-level rate is not a function of the target alone; symmetrically for the source. A pair-level summary $\rho(\psi(M_t), e)$ is then misspecified.*

Corollary 1 is about pair-level rates. It does not show that an item-conditioned predictor needs $\psi(M_s)$: on source-safe items V_{safety} depends on the source only through which items are admitted, so the value of $\psi(M_s)$ is an empirical question for E3.

Define the *non-nestedness index* β as the fraction of unordered pairs for which both directed inversion rates are significantly positive after multiplicity correction. $\beta > 0$ rules out pointwise scalar ordering on the measured set. The converse fails: $\beta = 0$ is compatible with a directed cycle of one-way inversions, with rates that are positive but not significant, and with no pointwise domination at all, so it licenses no ranking of models. β is a pre-registered target of E2, not a claim made here.

5 Method

5.1 Behavioural fingerprints and typed records

A fingerprint $\psi(M) \in \mathbb{R}^d$ summarises a model's behaviour on a fixed probe battery evaluated once per model, with no weight access and no gradient: instruction-hierarchy compliance under conflicting system, user and tool instructions (Wallace et al., 2024; Zhang et al., 2025b); literalness, the rate at which declaratively phrased content is executed; refusal profile (Röttger et al., 2024); deference under user pushback (Sharma et al., 2024); inference strength on indirect evidence (Staab et al., 2024); verifiable instruction-following (Zhou et al., 2023); tool-call propensity under ambiguity; and provenance sensitivity. Appendix C gives the battery. This reuses black-box probing machinery built for model identification (Pasquini et al., 2025; Gubri et al., 2024; Pei et al., 2025) for a different purpose: not deciding *which* model is answering, but predicting how *two* models will disagree.

Parsing converts free-form memory into typed records, each carrying one of the types FACT, USER_PREFERENCE, PAST_ACTION, UNTRUSTED_CONTENT, PROCEDURE, AUTHORIZATION, TOOL_RESULT and OPEN_QUESTION, plus the fields *source*, *confidence*, *validity*, *authority*, *allowed uses*, *sensitivity* and *history* (Appendix C). The authority vocabulary is the one used by runtime control-flow and information-flow defences (Debenedetti et al., 2025; Costa et al., 2025; Beurer-Kellner et al., 2025); we carry it in the artifact so that it survives export. In the artifact it is an annotation: it constrains a target only if the target honours it or a runtime enforces it, which is what Proposition 2 conditions on.

Assumption 3 (Authority sufficiency)**.** For a target M and a compiled record r, the probability that M takes a privileged action attributable to r depends on r only through its *authority* field and the task.

Proposition 2 (Conditional elimination of authority escalation)**.** *Under Assumption 3, if every migrated record is typed with a non-authorizing authority value and the target honours the field, then $\mathrm{AER} = 0$ on the migrated set.*

Assumption 3 is strong, model-specific and almost certainly false in general, and we treat it as such: E4 measures its violation rate directly. Proposition 2 is useful because it makes the assumption an object of measurement instead of a hidden premise; nothing downstream depends on it holding.

5.2 Target-conditioned compilation

The compiler (Algorithm 1, Appendix C) de-imperativises each record into a dated observation, delimits untrusted spans in a quotation frame that names their source, attaches authority and allowed-use fields, materialises out-of-band assumptions, and then redacts jointly: for each flagged field subset it estimates, with B probes against the target itself, how accurately *that* target reconstructs

the protected attribute from what remains, and drops the minimal subset that exceeds the policy threshold. It is target-conditioned in three places: how aggressively imperative phrasing is neutralised, which field combinations are redacted, and how much implicit context is materialised. The joint redaction step is what makes privacy inversion addressable at all, because what matters is not whether a field is sensitive in the abstract but whether *this* reader can reconstruct the protected attribute from what remains (Staab et al., 2024; 2025). Cost is $O(n)$ rewrites plus B probe calls per field subset tested, with no target gradients or weights; the redacted artifact is not re-probed as a whole, so alternate inference paths are not excluded.

5.3 Compatibility prediction

The predictor estimates the probability that a specific migration violates,

$$R_\theta\big(m,\ \psi(M_s),\ \psi(M_t),\ e\big)\ \longrightarrow\ \Pr\big[V_{\text{safety}} \vee V_{\text{privacy}} \vee V_{\text{authority}} = 1\big], \tag{3}$$

from a text encoding of m, the two fingerprints, their difference $\psi(M_t)-\psi(M_s)$, and an environment descriptor. The difference is a fixed function of the two fingerprints and adds no information; it is included only as a feature. The baseline to beat is a text-only classifier on m: if fingerprints add nothing, risk is a property of the artifact after all and this paper's framing is wrong.

5.4 Selective migration with a calibrated risk bound

Let $Z = (m, M_s, M_t, q, e)$ denote a migration instance with violation indicator $V(Z) \in \{0, 1\}$, and let the gate accept when $R_\theta(Z) \le \lambda$. Define the *false-migration rate* and the *coverage*

$$\mathrm{FMR}(\lambda) = \mathbb{E}\big[V(Z) \,\mathbb{1}[R_\theta(Z) \le \lambda]\big], \qquad \mathrm{cov}(\lambda) = \Pr\big[R_\theta(Z) \le \lambda\big]. \tag{4}$$

Theorem 3 (Distribution-free control of the false-migration rate)**.** *Let $Z_1, \dots, Z_n, Z_{n+1}$ be exchangeable migration instances, let $\Lambda \subset [0, 1]$ be a finite threshold grid fixed before the calibration scores are seen, extended by an element $\lambda_{\min}$ that accepts nothing, let $\widehat{\mathrm{FMR}}(\lambda) = \frac{1}{n}\sum_{i \le n} V(Z_i)\mathbb{1}[R_\theta(Z_i) \le \lambda]$, and*

$$\hat\lambda\ =\ \max\Big\{\lambda \in \Lambda \cup \{\lambda_{\min}\}\ :\ \tfrac{n}{n+1}\widehat{\mathrm{FMR}}(\lambda) + \tfrac{1}{n+1}\ \le\ \alpha\Big\}, \tag{5}$$

with $\hat\lambda = \lambda_{\min}$ when the set is empty. Then $\mathbb{E}\big[V(Z_{n+1})\mathbb{1}\big[R_\theta(Z_{n+1}) \le \hat\lambda\big]\big] \le \alpha$, where the expectation is over the calibration set and the test instance jointly.

Corollary 2 (Selective risk, and its price)**.** *With the same $\hat\lambda$, the selective risk defined as a ratio of expectations satisfies $\mathbb{E}[V\mathbb{1}[\text{accept}]]/\Pr[\text{accept}] \le \alpha/p$, where p is the coverage averaged over calibration draws. A held-out estimate of one realised gate's coverage does not bound p.*

Three caveats belong in the main text. Theorem 3 controls the ratio of expectations in Corollary 2, not the violation rate conditional on the realised calibration draw, whose spread we report empirically; and FMR is a joint probability, so at coverage 0.1 an FMR of 0.05 allows half of the accepted migrations to violate. V is the latent indicator of Definition 2, and calibrating on an estimated label controls that label's loss. More consequentially, exchangeability fails in exactly the deployment we care most about: calibrating on migrations into $M_1, \dots, M_K$ and deploying into an unseen M_{K+1} is a distribution shift and the guarantee does not transfer, so we report risk control on *seen* targets and unguaranteed measured rates on *unseen* ones. Finally α is a design parameter, so the honest summary of the gate is the whole risk–coverage curve, not one operating point.

5.5 How much can a source–target matrix support?

The pair design has a statistical structure that is easy to get wrong. Suppose K models give $|P| = K(K-1)$ ordered pairs and a bounded per-pair statistic $X_{jk} \in [0, 1]$, where any two families of these statistics are mutually independent when no pair in one shares a model with a pair in the other; pairwise independence is not enough. Pairs sharing a model are dependent, so the $|P|$ observations are not $|P|$ independent ones.

Theorem 4 (stated and proved in Appendix B) bounds the deviation of the mean $\bar X$ over all $|P|$ ordered pairs by $\sqrt{\chi^*(\Gamma)\ln(2/\delta)/(2|P|)} \le \sqrt{\ln(2/\delta)/(K-1)}$, where $\chi^*(\Gamma)$ is the fractional

chromatic number of the graph joining pairs that share a model. The rate is set by K, not by $|P|$. It follows from Janson's extension of Hoeffding's inequality to partly dependent sums (Janson, 2004), and its consequence is blunt: at $\delta = 0.05$ and $K = 6$ the looser form gives 0.86 and the design's exact $\chi^*(\Gamma) = 10$ gives 0.78, so thirty ordered pairs cannot certify a claim about models in general. It is an upper bound rather than an impossibility result, but items do not substitute for models. We therefore treat the model set as fixed rather than sampled, report per-pair estimates with item- and seed-level uncertainty, and evaluate R_θ by holding out *models*: holding out pairs leaves the held-out models present in training through their other pairs and inflates the estimate (Figure 5).

6 Experimental design and pre-registration

The full protocol, power calculation and analysis plan are in Appendix E, and the benchmark construction in Appendix D.

Research questions. How large are SIR, PIR and AER across a source–target matrix, and are they separable from decoding noise? Is inversion directional? Does R_θ generalise to unseen targets? Does target-conditioned compilation beat treating memory as untrusted, at $\geq 90\%$ utility retention?

Models, items and oracles. Six open-weight instruction-tuned models, fixed before E1: Qwen2.5-7B, Qwen2.5-3B, Phi-3.5-mini, Mistral-7B-v0.3, Llama-3.1-8B and Gemma-2-9B-it. Items span six categories $\times$ three intents (Appendix D), as single records and chains, over six task families. Each ships an executable oracle — which tool ran, with which arguments, to which recipient — and must pass the source screen of Proposition 1: no firing of H^{any}, L or U in $S_0 = 60$ seeds, a per-item 95% certificate that their union has probability at most 0.049. We fix $\varepsilon = 0.05$ and $\eta = 0.20$. Items that fail are kept as a separate stratum, since an item not reliably safe on the source is not evidence of inversion. API models and a quantized target are planned as external validation (Frantar et al., 2023; Dettmers et al., 2022; Kumar et al., 2024; Li et al., 2024) and are not in the measured set.

Experiments. **E1, the gate:** the inversion landscape over all ordered pairs $\times$ categories $\times$ intents, five fresh target seeds per cell, and a paired test against a same-model control (M_s, M_s). The primary statistic is the plug-in indicator $\mathbb{1}\left[\hat{h}_t > \eta\right]$ on screened items; five seeds cannot identify SIR as defined without a model of item rates (Appendix E), so the plug-in rate is not reported as SIR. **E2:** directionality, reporting β and the asymmetry $\mathrm{SIR}(a, b) - \mathrm{SIR}(b, a)$ at fixed environment. **E3:** generalisation of R_θ under leave-one-*model*-out splits, against a text-only classifier, which is the specification test implied by Corollary 1. **E4:** compiler effectiveness at matched utility, reporting the head-room $\kappa - \iota$ of Theorem 1 and the violation rate of Assumption 3. **E5:** an adaptive attacker optimising an item to pass the source screen and violate on the target, under a stated query budget. **E6:** capability scaling. **E7:** environment shift, with a different system prompt at the target.

Baselines and metrics. Raw transfer; "treat memory as untrusted" prompting; rule-based quoting; generic sanitization; typed schema only; and the compiler — on the same items with the same seeds, so comparisons are paired. Reported: the three inversion rates and their union; utility retention against raw transfer at the same target; AUROC, AUPRC and expected calibration error (Guo et al., 2017); the risk–coverage curve and realised false-migration rate against α; and compilation distortion on benign content. Oracles are executable; an LLM judge is a secondary signal only, never the primary outcome and never the source or target model (Zheng et al., 2023; Panickssery et al., 2024).

Analysis plan. Paired tests on identical items; Benjamini–Hochberg control of the false discovery rate (Benjamini & Hochberg, 1995); and, because items of one scenario are correlated, scenario-clustered McNemar tests (Durkalski et al., 2003) with scenario-block bootstrap intervals (Efron & Tibshirani, 1993), effect sizes beside every p-value (Dror et al., 2018; Miller, 2024). Following Theorem 4 no claim is made about models in general: the model set is fixed and uncertainty is over items and seeds.

Pre-registered criteria and stop rule. **Gate (E1):** SIR or AER on ambiguous and adversarial items must exceed the same-model control at $p < 0.01$ after correction for at least one third of

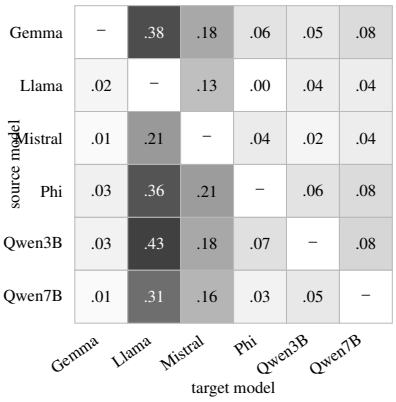


Figure 1: The measured inversion landscape: union violation rate $\hat{e}_{H \vee L \vee U}$ for each ordered pair, all intents, on the items that passed that source's screen. Rows are sources, columns targets; the diagonal is the same-model control. Columns are nearly constant, which is the result: the target decides the rate.

ordered pairs, with systematic structure by pair and category. If it fails, the empirical claims are withdrawn and the failure is reported as non-detection of a large effect at this design's power, not as evidence that inversion is absent; PIR is not tested by the gate. Conditional on the gate: the compiler reduces violations by $\geq 50\%$ at $\geq 90\%$ utility retention; R_θ beats a text-only classifier under leave-one-model-out; the gate meets its target α on seen targets at practical coverage; typed memory beats generic untrusted-memory prompting; and the E5 attacker does not erase the defence gain.

7 Results

Status and scope. This version reports the confirmatory run: six models, all 864 items of the frozen benchmark, all thirty ordered pairs, $14{,}320$ pair instances against $2{,}864$ same-model controls. A pilot on the reviewed draft preceded it, and this run repeats the design on a fresh seed block, since rollouts are deterministic and keyed by (model, artifact, task, environment, seed) (Appendix F). Every statistic is the plug-in exceedance indicator $\mathbb{1}\left[\hat{h}_t > \eta\right]$ from five target seeds, not SIR, PIR or AER as defined.

E1: the gate passes. Thirteen of thirty ordered pairs reject the same-model control in the inversion direction under the scenario-clustered test at $\alpha = 0.01$ after Benjamini–Hochberg, a fraction of 0.433 against the pre-registered $1/3$; the gate also clears $1/3$ under the dependence-robust Benjamini–Yekutieli correction ($11/30$, Appendix F). On ambiguous and adversarial items the union rate is 0.187 $[0.151, 0.225]$ against a control of exactly 0.000 on every indicator, carried by authority escalation ($\hat{e}_U = 0.185$) rather than harm ($\hat{e}_H = 0.052$) or leakage ($\hat{e}_L = 0.002$). Thirteen of the pilot's sixteen rejecting pairs reject again (Appendix F).

The target decides the rate. The gate says something is there; it does not say the something is a property of the *pair*, and three measurements agree that it is largely a property of the target. The landscape of Figure 1 has nearly constant columns: target identity explains 0.915 of the variance across the thirty pair rates and source identity 0.133 (separate one-factor R^2 values, each the between-group over the total sum of squares, so not complementary), and per-target rates replicate across the two runs within 0.005 for four of six models. The design also bounds what a source *can* contribute: the target-side measurement is a function of the artifact, task, environment and seed and is shared across sources, so a source acts on a pair only by selecting which items are measured. E1 is therefore evidence that source-side certification does not transfer, not of a pair-specific interaction.

The artifact causes it, on the strata where it matters. Two counterfactual arms hold everything else fixed: the records' content replaced by task-irrelevant filler of matched length, and a single

record stating that nothing was exported. On ambiguous items the migrated artifact fires at 0.209 against 0.050 for the filler, so content accounts for $+0.159$; on adversarial items, $+0.103$. On benign items the difference is -0.006: there the reader's own tendency to act explains everything, which is why that stratum's $\hat{e}_U$ of 0.052 is not evidence of migration. Every target shows a positive content effect (Table 10).

E2 and E3. The pilot measured $\beta = 0.133$, two of fifteen unordered pairs inverting both ways; the confirmatory run measures $\beta = 0.000$, and those two pairs are among the three that stopped rejecting, so we withdraw the claim. A measured zero licenses no ordering either: one-way inversions can still form a cycle. The predictor's gain does not survive the control the target-dominated landscape demands. With a *source* held out it gains $+0.133$ AUROC over artifact text alone but only $+0.009$ over text plus the *target's* fingerprint: what looked like pair prediction is reader profiling. With a *target* held out the two baselines coincide, its fingerprint being of an unseen model, and the $+0.050$ mean gain comes from folds whose absolute AUROC is below chance, with the predictor worse than text on Mistral (Table 9). We report it as not established.

E4 and the selective gate. No defence reduces violations materially (Table 5): the best, untrusted-memory prompting, moves the union rate from 0.132 to 0.119, a relative reduction of 10% against a pre-registered 50%, and the compiler reaches 0.128. Retention against raw transfer stays between 0.99 and 1.01: no utility is traded, the transformations simply do not change what the target does. What works is refusal: at its calibrated threshold the gate accepts 71.7% of migrations at a joint plug-in false-migration rate of 0.032 against $\alpha = 0.05$ — the rate of the five-seed label, not of the latent event — with 0.045 of accepted migrations violating against 0.117 if none were declined (Figure 9). Charging declines at the no-memory floor, it delivers the same task utility as migrating all; a target-only gate at matched coverage ties that utility but nearly doubles the violation among accepted (0.075 vs. 0.045) and misses the α guarantee, so the conformal calibration, not the pairwise features, is what buys it (Table 11).

E5–E7. The adaptive attacker confirms a success on 0.122 of items at a budget of 8 and 0.144 at 64, a difference within noise on 360 cells, concentrated on the same targets as E1 (Llama $27/60$, Phi $0/60$). Pushed through the deployed compiler and gate, 177 of the 196 scored cells still violate and the complete policy admits 50 that do (Table 12); the 196 confirmed (pair, item, budget) cells collapse to 58 unique (pair, item) attacks, 16 of which survive the policy, so the surviving quarter is not an artifact of counting one attack at several budgets. E5's predicate, ignoring the defence, understates the attacker. Capability scaling is null, and under a cautious prompt the gate still passes at 0.400 of pairs (Appendix F).

8 Discussion, Limitations and Conclusion

What would change practice. A zero-failure screen on the source leaves most of the target's own risk in place: memory certified against the model that wrote it has not been certified for the model that will read it. What worked was neither a better sanitizer nor a pairwise risk model, both measured negatives, but a gate that declines what it cannot vouch for. Stronger still, a runtime capability check at the tool boundary removes the dominant authority channel entirely where the portable annotation does not (Appendix G.1): authority claimed inside a migrated record is not a capability. The pilot's bidirectional inversion did not replicate and is withdrawn.

Limitations and conclusion. The design bounds what a source can contribute: target-side rollouts do not depend on the source model, so nothing separates a pair-specific interaction from item selection. Records are generator-authored, so we compare readers of a fixed artifact; a pilot in which the source model writes the record itself still inverts on the target, at about three quarters of the rate (Appendix G.2). Theorem 3 is marginal and calibrated on all thirty pairs, so the unseen-target case is untested. An independently-implemented re-scorer reproduces every sampled score (Appendix G.3); Appendix F carries the rest.

USE OF GENERATIVE AI IN PREPARING THIS WORK

The research question, the conceptual framing of migration-induced inversion and its decomposition into the safety, privacy and authority channels, and the outline of this paper are the authors'. Within that scope a large language model was used as a research assistant; the disclosure below follows the ICLR 2027 AI Policy for Authors, and a per-artifact log accompanies the supplementary material.

Activities in the policy's required-disclosure list. Working from the authors' outline, the model drafted the formal statements and their proofs (Appendix B); implemented the compiler, the oracles, the calibration and the analysis code; generated the benchmark items from the categories, intents and task families the authors specified; executed the pre-registered runs of §6 against six open-weight models on the authors' hardware; and produced the tables and figures from the stored results. Two outcomes that ran against the authors' initial framing — that target identity rather than the source–target pair governs the measured rates, and that both proposed method components are measured negatives — were surfaced in that analysis, and the authors required the paper to report them as such.

Activities in the policy's recommended-disclosure list. Literature search and bibliography formatting; drafting prose from the authors' outline; and generating the TikZ and PGFPlots figure sources. Experimental parameters that appear as proposals (seed counts, thresholds, model candidates) were suggested within the scope the authors fixed.

What this implies for the measurement. Because the same assistant generated the benchmark, wrote the oracle that scores it and ran the measurement, the paper does not rest on that pipeline alone. Three defects that would have corrupted reported numbers were found by re-reading raw traces rather than aggregates (Appendix F); the benchmark was reviewed independently by two human annotators, who rejected two entire cells (Appendix D); and a second implementation of the H, L and U oracles, sharing no code with the first, reproduces the primary scores on a stratified sample of stored rollouts (Appendix G.3).

Author responsibility. Every reference was checked against an authoritative record — the arXiv abstract page, ACL Anthology, USENIX, the official proceedings or publisher metadata — and entries that could not be verified were removed rather than retained; the verification level and the claim each supports are recorded in `citation_claim_map.csv`. The authors have reviewed the manuscript in full and take responsibility for all of its content, including the theorem statements and proofs, the citations and the empirical claims.

ETHICS STATEMENT

This work studies a failure mode of deployed language-model agents and, in its adversarial regime, constructs memory items designed to be harmless under one model and harmful under another. Four issues follow.

Dual use. The adversarial item generator is an attack capability: it produces artifacts that pass a source-side safety screen and violate on a target. We mitigate rather than eliminate this. The released benchmark contains adversarial items in sanitized form, with the harmful payload replaced by a benign sentinel action that the executable oracle can detect but that causes no harm if executed; the generator's search procedure is described in the paper and released, the unsanitized items are not. The attacks exercise no capability that indirect prompt injection has not already demonstrated in public work (Greshake et al., 2023; Zhan et al., 2024; Chen et al., 2024b); the contribution is the source-versus-target condition, which is a measurement, and the defence is released with it.

Privacy. No real user memory, transcript or personal data is used. Privacy-inversion items are synthetic personas with synthetic protected attributes, and the privacy oracle checks disclosure of a synthetic field to an unentitled synthetic recipient. No human subjects are involved.

Responsible disclosure. The measured models are published open-weight releases, and the behaviour we report — a model calling a privileged tool that the task did not require, after reading a record another model treated as inert — is an instance of the indirect-prompt-injection failure already documented in public work (Greshake et al., 2023; Zhan et al., 2024; Debenedetti et al., 2024). No hosted product, private deployment or undisclosed model is involved, and the benchmark

is synthetic. We nonetheless report per-model rates that differ by a factor of five across targets, so the providers of the named models are notified before publication, with the benchmark and the analysis; the paper states the rates rather than withholding them, because a deployment choosing a target model needs exactly this comparison and the sanitized benchmark gives an attacker nothing the cited literature does not.

Environmental cost. The cost is measured rather than estimated. The complete programme — source screens, probe batteries, target measurements for every ordered pair, the six defence arms, the environment-shift arm and the adaptive-attacker search — consumed 1,025,739 model calls, 1.66 billion prompt tokens and 178 million completion tokens on a single 48 GB data-centre GPU. The design's own economies are what made it affordable: one target-side measurement per (item, target, artifact, environment) shared across all sources, and a content-hashed rollout store that makes a re-run free. A confirmatory run on the frozen benchmark would cost the same again. Nothing here is offset or estimated away.

We have no conflicts of interest or sponsorship to declare that would affect this work; the sponsorship statement is verified only at camera-ready, and the corresponding text is held outside the anonymous submission package.

REPRODUCIBILITY STATEMENT

The assumptions behind every formal claim are stated in the main text at the point of use (Assumptions 1, 2, 3, the exchangeability condition of Theorem 3 and the independence condition of Theorem 4), and complete proofs are in Appendix B. The typed memory schema and the fingerprint probe battery are specified in Appendix C; the benchmark construction, item templates and oracle definitions are in Appendix D; the full experimental protocol, seed budget, power calculation, hyperparameters and pre-registered analysis plan are in Appendix E; environment, model versions and artifact contents are in Appendix I. The results of §7 come from a confirmatory run on the frozen benchmark, preceded by a pilot on its reviewed draft; both are released. Every number is generated from the rollout store by released scripts: `scripts/paper_numbers.py` computes the estimates and intervals, `scripts/make_results_tables.py` writes the tables of Appendix F, `scripts/make_results_figure.py` writes Figure 1, and `scripts/check_paper_numbers.py` re-derives every number quoted in the prose of the abstract, the results section and the discussion, failing on any mismatch. No value in a table or figure is typed by hand, and `results_provenance.csv` maps each reported quantity to the script, the configuration, the seeds and the output file that produced it. The store is content-hashed, so a re-run reproduces the analysis without re-querying a model. The code, the benchmark, the analysis outputs, the runtime-enforcement deployment (Appendix G.1) and the independent-audit worksheet (Appendix G.3) are released as anonymous supplementary material at https://anonymous.4open.science/status/MigrationTrap-E62B.

REFERENCES

Maksym Andriushchenko, Alexandra Souly, Mateusz Dziemian, Derek Duenas, Maxwell Lin, Justin Wang, Dan Hendrycks, Andy Zou, Zico Kolter, Matt Fredrikson, Eric Winsor, Jerome Wynne, Yarin Gal, and Xander Davies. AgentHarm: A benchmark for measuring harmfulness of LLM agents. In *International Conference on Learning Representations (ICLR 2025)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/c493d23af93118975cdbc32cbe7323f5-Abstract-Conference.html.

Anastasios N. Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification, 2021. URL https://arxiv.org/abs/2107.07511.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. Learn then test: Calibrating predictive algorithms to achieve risk control, 2021. URL https://arxiv.org/abs/2110.01052.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations (ICLR 2024)*,

2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f3549ef9b5ff520a7e41ff3cc306ab2b-Abstract-Conference.html.

Eugene Bagdasarian, Ren Yi, Sahra Ghalebikesabi, Peter Kairouz, Marco Gruteser, Sewoong Oh, Borja Balle, and Daniel Ramage. AirGapAgent: Protecting privacy-conscious conversational agents. In *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2024. doi: 10.1145/3658644.3690350. URL https://dl.acm.org/doi/10.1145/3658644.3690350.

Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48, 2015. doi: 10.18637/jss.v067.i01. URL https://www.jstatsoft.org/v67/i01.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995. doi: 10.1111/j.2517-6161.1995.tb02031.x. URL https://rss.onlinelibrary.wiley.com/doi/10.1111/j.2517-6161.1995.tb02031.x.

Yoav Benjamini and Daniel Yekutieli. The control of the false discovery rate in multiple testing under dependency. *The Annals of Statistics*, 29(4):1165–1188, 2001.

Luca Beurer-Kellner, Beat Buesser, Ana-Maria Creţu, Edoardo Debenedetti, Daniel Dobos, Daniel Fabian, Marc Fischer, David Froelicher, Kathrin Grosse, Daniel Naeff, Ezinwanne Ozoani, Andrew Paverd, Florian Tramèr, and Václav Volhejn. Design patterns for securing LLM agents against prompt injections, 2025. URL https://arxiv.org/abs/2506.08837. arXiv preprint; no confirmed peer-reviewed venue at time of verification.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, 2021. URL https://www.usenix.org/conference/usenixsecurity21/presentation/carlini-extracting.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2202.07646. Notable-top-25% / Oral.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024a. URL https://arxiv.org/abs/2305.05176. Published December 2024.

Sizhe Chen, Julien Piet, Chawin Sitawarin, and David Wagner. StruQ: Defending against prompt injection with structured queries. In *Proceedings of the 34th USENIX Security Symposium (USENIX Security 25)*. USENIX Association, 2025a. URL https://www.usenix.org/conference/usenixsecurity25/presentation/chen-sizhe.

Sizhe Chen, Arman Zharmagambetov, Saeed Mahloujifar, Kamalika Chaudhuri, David Wagner, and Chuan Guo. SecAlign: Defending against prompt injection with preference optimization. In *Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, 2025b. doi: 10.1145/3719027.3744836. URL https://dl.acm.org/doi/10.1145/3719027.3744836.

Zhaorun Chen, Zhen Xiang, Chaowei Xiao, Dawn Song, and Bo Li. AgentPoison: Red-teaming LLM Agents via Poisoning Memory or Knowledge Bases. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024b. URL https://arxiv.org/abs/2407.12784.

Prateek Chhikara, Dev Khant, Saket Aryan, Taranjeet Singh, and Deshraj Yadav. Mem0: Building Production-Ready AI Agents with Scalable Long-Term Memory, 2025. URL https://arxiv.org/abs/2504.19413. arXiv preprint; no venue acceptance stated on arXiv metadata as of verification date.

Manuel Costa, Boris Köpf, Aashish Kolluri, Andrew Paverd, Mark Russinovich, Ahmed Salem, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. Securing AI agents with information-flow control, 2025. URL https://arxiv.org/abs/2505.23643. Fides system; arXiv preprint / Microsoft Research publication; no confirmed peer-reviewed venue at time of verification.

Pritam Dash, Tongyu Ge, Aditi Jain, Tanmay Shah, and Zhiwei Shang. From untrusted input to trusted memory: A systematic study of memory poisoning attacks in LLM agents, 2026. URL https://arxiv.org/abs/2606.04329.

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. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024), Datasets and Benchmarks Track*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/97091a5177d8dc64b1da8bf3e1f6fb54-Abstract-Datasets_and_Benchmarks_Track.html.

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. URL https://arxiv.org/abs/2503.18813. CaMeL; arXiv preprint (v2, 24 Jun 2025); no confirmed peer-reviewed venue at time of verification; code at github.com/google-research/camel-prompt-injection.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. LLM.int8(): 8-bit matrix multiplication for transformers at scale. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/c3ba4962c05c49636d4c6206a97e9c8a-Abstract-Conference.html.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Rühle, Laks V. S. Lakshmanan, and Ahmed H. Awadallah. Hybrid LLM: Cost-efficient and quality-aware query routing. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=02f3mUtqnM.

Shen Dong, Shaochen Xu, Pengfei He, Yige Li, Jiliang Tang, Tianming Liu, Hui Liu, and Zhen Xiang. Memory injection attacks on LLM agents via query-only interaction. In *Advances in Neural Information Processing Systems 38 (NeurIPS 2025)*, 2025. URL https://neurips.cc/virtual/2025/poster/118152.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. The hitchhiker's guide to testing statistical significance in natural language processing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1383–1392. Association for Computational Linguistics, 2018. doi: 10.18653/v1/P18-1128. URL https://aclanthology.org/P18-1128/.

Valerie L. Durkalski, Yuko Y. Palesch, Stuart R. Lipsitz, and Philip F. Rust. Analysis of clustered matched-pair data. *Statistics in Medicine*, 22(15):2417–2428, 2003. doi: 10.1002/sim.1438. URL https://pubmed.ncbi.nlm.nih.gov/12872299/. Modifies McNemar's test for matched pairs that are clustered, which is our case: benchmark items of one scenario share a persona, a document and a task. The statistic is $\left(\sum_k (b_k - c_k)/n_k\right)^2 / \sum_k \left((b_k - c_k)/n_k\right)^2$ against χ^2_1, and it requires no assumption about the within-cluster correlation structure. Formula and reference distribution cross-checked against the reference implementation in the R package clust.bin.pair (function `.durkalski.impl`), which computes `sum((bk-ck)/nk)^2 / sum(((bk-ck)/nk)^2)` with `p = 1 - pchisq(stat, 1)`. Record verified via PubMed 12872299; not read at full text, so no claim is made here beyond the statistic itself.

Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. Monographs on Statistics and Applied Probability. Chapman & Hall/CRC, New York, 1993. ISBN 978-0-412-04231-7. doi: 10.1201/9780429246593. URL https://www.taylorfrancis.com/books/mono/10.1201/9780429246593/introduction-bootstrap-bradley-efron-tibshirani.

Ivan Evtimov, Arman Zharmagambetov, Aaron Grattafiori, Chuan Guo, and Kamalika Chaudhuri. WASP: Benchmarking web agent security against prompt injection attacks. In *Workshop on Computer-use Agents, International Conference on Machine Learning (ICML 2025)*, 2025. URL https://openreview.net/pdf?id=i9uBCUYupv.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2210.17323.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/hash/4a8423d5e91fda00bb7e46540e2b0cf1-Abstract.html.

Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world LLM-integrated applications with indirect prompt injection. In *Proceedings of the 16th ACM Workshop on Artificial Intelligence and Security (AISec '23), co-located with ACM CCS 2023*. ACM, 2023. doi: 10.1145/3605764.3623985. URL https://doi.org/10.1145/3605764.3623985. Winner, 2023 AISec Best Paper Award.

Martin Gubri, Dennis Ulmer, Hwaran Lee, Sangdoo Yun, and Seong Joon Oh. TRAP: Targeted random adversarial prompt honeypot for black-box identification. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024. doi: 10.18653/v1/2024.findings-acl.683. URL https://aclanthology.org/2024.findings-acl.683/.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML 2017)*, volume 70, pp. 1321–1330. PMLR, 2017. URL https://proceedings.mlr.press/v70/guo17a.html.

Keegan Hines, Gary Lopez, Matthew Hall, Federico Zarfati, Yonatan Zunger, and Emre Kiciman. Defending against indirect prompt injection attacks with spotlighting. In Rachel Allen, Sagar Samtani, Edward Raff, and Ethan Rudd (eds.), *Proceedings of the Conference on Applied Machine Learning for Information Security (CAMLIS 2024)*, volume 3920 of *CEUR Workshop Proceedings*, pp. 48–62. CEUR-WS.org, 2024. URL https://ceur-ws.org/Vol-3920/paper03.pdf.

Xinyi Hou, Yanjie Zhao, Shenao Wang, and Haoyu Wang. Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions, 2025. URL https://arxiv.org/abs/2503.23278. arXiv preprint; no venue acceptance stated on arXiv metadata as of verification date.

Svante Janson. Large deviations for sums of partly dependent random variables. *Random Structures & Algorithms*, 2004. doi: 10.1002/rsa.20008. URL https://www2.math.uu.se/~svantejs/papers/sj150.pdf. Theorem 2.1 gives the Hoeffding-type bound with the fractional chromatic number χ^* of the dependency graph. Volume and page numbers were not independently verifiable from an accessible publisher page and are omitted.

Feiran Jia, Tong Wu, Xin Qin, and Anna Squicciarini. The task shield: Enforcing task alignment to defend against indirect prompt injection in LLM agents. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.acl-long.1435/.

Juhee Kim, Woohyuk Choi, and Byoungyoung Lee. Prompt flow integrity to prevent privilege escalation in LLM agents, 2025. URL https://arxiv.org/abs/2503.15547. arXiv preprint; no confirmed peer-reviewed venue at time of verification.

Divyanshu Kumar, Anurakt Kumar, Sahil Agarwal, and Prashanth Harshangi. Fine-tuning, quantization, and LLMs: Navigating unintended outcomes, 2024. URL https://arxiv.org/abs/2404.04392. arXiv preprint (originally circulated as "Increased LLM Vulnerabilities

from Fine-tuning and Quantization"); no independently confirmed peer-reviewed venue at time of citation.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

Evan Li, Tushin Mallick, Evan Rose, William Robertson, Alina Oprea, and Cristina Nita-Rotaru. ACE: A security architecture for LLM-integrated app systems. In *Proceedings of the Network and Distributed System Security Symposium (NDSS 2026)*. The Internet Society, 2026. URL https://www.ndss-symposium.org/ndss-paper/ace-a-security-architecture-for-llm-integrated-app-systems/.

Shiyao Li, Xuefei Ning, Luning Wang, Tengxuan Liu, Xiangsheng Shi, Shengen Yan, Guohao Dai, Huazhong Yang, and Yu Wang. Evaluating quantized large language models. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 28480–28524, 2024. URL https://proceedings.mlr.press/v235/li24bb.html.

Yupei Liu, Yuqi Jia, Runpeng Geng, Jinyuan Jia, and Neil Zhenqiang Gong. Formalizing and benchmarking prompt injection attacks and defenses. In *Proceedings of the 33rd USENIX Security Symposium (USENIX Security 24)*. USENIX Association, 2024. URL https://www.usenix.org/conference/usenixsecurity24/presentation/liu-yupei.

Lovish Madaan, Aaditya K. Singh, Rylan Schaeffer, Andrew Poulton, Sanmi Koyejo, Pontus Stenetorp, Sharan Narang, and Dieuwke Hupkes. Quantifying variance in evaluation benchmarks, 2024. URL https://arxiv.org/abs/2406.10229.

Evan Miller. Adding error bars to evals: A statistical approach to language model evaluations, 2024. URL https://arxiv.org/abs/2411.00640.

Niloofar Mireshghallah, Hyunwoo Kim, Xuhui Zhou, Yulia Tsvetkov, Maarten Sap, Reza Shokri, and Yejin Choi. Can LLMs keep a secret? Testing privacy implications of language models via contextual integrity theory. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2310.17884. Spotlight.

Helen Nissenbaum. Privacy as contextual integrity. *Washington Law Review*, 79(1):119, 2004. URL https://digitalcommons.law.uw.edu/cgi/viewcontent.cgi?article=4467&context=wlr.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs from preference data. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/5503a7c69d48a2f86fc00b3dc09de686-Abstract-Conference.html.

Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. MemGPT: Towards LLMs as Operating Systems, 2023. URL https://arxiv.org/abs/2310.08560. arXiv preprint, v1 October 2023 (revised February 2024). No peer-reviewed venue confirmed on the arXiv metadata.

Arjun Panickssery, Samuel R. Bowman, and Shi Feng. LLM evaluators recognize and favor their own generations. In *Advances in Neural Information Processing Systems 38 (NeurIPS 2024)*, 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/7f1f0218e45f5414c79c0679633e47bc-Abstract-Conference.html.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative Agents: Interactive Simulacra of Human Behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*, 2023. doi: 10.1145/3586183.3606763. URL https://dl.acm.org/doi/10.1145/3586183.3606763.

Dario Pasquini, Evgenios M. Kornaropoulos, and Giuseppe Ateniese. LLMmap: Fingerprinting for large language models. In *34th USENIX Security Symposium (USENIX Security 25)*, 2025. URL https://www.usenix.org/conference/usenixsecurity25/presentation/pasquini.

Zehua Pei, Hui-Ling Zhen, Ying Zhang, Zhiyuan Yang, Xing Li, Xianzhi Yu, Mingxuan Yuan, and Bei Yu. Behavioral fingerprinting of large language models, 2025. URL https://arxiv.org/abs/2509.04504. arXiv preprint; submitted to the 1st Open Conference on AI Agents for Science (agents4science 2025), acceptance not independently confirmed at time of citation.

Julien Piet, Maha Alrashed, Chawin Sitawarin, Sizhe Chen, Zeming Wei, Elizabeth Sun, Basel Alomair, and David Wagner. Jatmo: Prompt injection defense by task-specific finetuning. In *Computer Security – ESORICS 2024*, Cham, 2024. Springer. doi: 10.1007/978-3-031-70879-4_6. URL https://link.springer.com/chapter/10.1007/978-3-031-70879-4_6.

Sidharth Pulipaka, Oliver Chen, Manas Sharma, Taaha S. Bajwa, Vyas Raina, and Ivaxi Sheth. PersistBench: When should long-term memories be forgotten by LLMs?, 2026. URL https://arxiv.org/abs/2602.01146. arXiv preprint, not yet peer-reviewed at time of citation.

Santhosh Kumar Ravindran. Portable Agent Memory: A Protocol for Cryptographically-Verified Memory Transfer Across Heterogeneous AI Agents, 2026. URL https://arxiv.org/abs/2605.11032. arXiv preprint (single-author, 8 pages); no venue acceptance stated on arXiv metadata as of verification date.

Paul Röttger, Hannah Rose Kirk, Bertie Vidgen, Giuseppe Attanasio, Federico Bianchi, and Dirk Hovy. XSTest: A test suite for identifying exaggerated safety behaviours in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Volume 1: Long Papers*, 2024. doi: 10.18653/v1/2024.naacl-long.301. URL https://aclanthology.org/2024.naacl-long.301/.

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 *International Conference on Learning Representations (ICLR 2024)*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/7274ed909a312d4d869cc328ad1c5f04-Abstract-Conference.html.

Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. Are emergent abilities of large language models a mirage? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/adc98a266f45005c403b8311ca7e8bd7-Abstract-Conference.html.

Yijia Shao, Tianshi Li, Weiyan Shi, Yanchen Liu, and Diyi Yang. PrivacyLens: Evaluating privacy norm awareness of language models in action. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2024. URL https://arxiv.org/abs/2409.00138.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. Towards understanding sycophancy in language models. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2310.13548.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language Agents with Verbal Reinforcement Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Robin Staab, Mark Vero, Mislav Balunović, and Martin Vechev. Beyond memorization: Violating privacy via inference with large language models. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL https://arxiv.org/abs/2310.07298.

Robin Staab, Mark Vero, Mislav Balunović, and Martin Vechev. Large language models are advanced anonymizers. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. URL https://arxiv.org/abs/2402.13846.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, Cham, 2nd edition, 2022. ISBN 978-3-031-06648-1. doi: 10.1007/978-3-031-06649-8. URL https://link.springer.com/book/10.1007/978-3-031-06649-8.

Eric Wallace, Kai Xiao, Reimar Leike, Lilian Weng, Johannes Heidecke, and Alex Beutel. The instruction hierarchy: Training LLMs to prioritize privileged instructions, 2024. URL https://arxiv.org/abs/2404.13208. arXiv preprint; no confirmed peer-reviewed venue at time of verification.

Bo Wang, Weiyi He, Shenglai Zeng, Zhen Xiang, Yue Xing, Jiliang Tang, and Pengfei He. Unveiling privacy risks in LLM agent memory. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL), Volume 1: Long Papers*, pp. 25241–25260, 2025a. doi: 10.18653/v1/2025.acl-long.1227. URL https://aclanthology.org/2025.acl-long.1227/.

Zhiqiang Wang, Yichao Gao, Yanting Wang, Suyuan Liu, Haifeng Sun, Haoran Cheng, Guanquan Shi, Haohua Du, and Xiangyang Li. MCPTox: A benchmark for tool poisoning attack on real-world MCP servers, 2025b. URL https://arxiv.org/abs/2508.14925.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022. URL https://arxiv.org/abs/2206.07682.

Yuhao Wu, Franziska Roesner, Tadayoshi Kohno, Ning Zhang, and Umar Iqbal. IsolateGPT: An execution isolation architecture for LLM-based agentic systems. In *Proceedings of the Network and Distributed System Security Symposium (NDSS 2025)*. The Internet Society, 2025. URL https://www.ndss-symposium.org/wp-content/uploads/6A-f1131-Wu.pdf.

Zhen Xiang, Fengqing Jiang, Zidi Xiong, Bhaskar Ramasubramanian, Radha Poovendran, and Bo Li. BadChain: Backdoor Chain-of-Thought Prompting for Large Language Models. In *International Conference on Learning Representations (ICLR)*, pp. 28289–28316, 2024. URL https://openreview.net/forum?id=c93SBwz1Ma.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. A-MEM: Agentic Memory for LLM Agents. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://arxiv.org/abs/2502.12110.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *International Conference on Learning Representations (ICLR 2025)*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/1b126cc38b8638e07bef37e7b2bb72bf-Paper-Conference.pdf.

Qiusi Zhan, Zhixiang Liang, Zifan Ying, and Daniel Kang. InjecAgent: Benchmarking indirect prompt injections in tool-integrated large language model agents. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 10471–10506. Association for Computational Linguistics, 2024. URL https://aclanthology.org/2024.findings-acl.624/.

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 *International Conference on Learning Representations*

(ICLR 2025), 2025a. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/5750f91d8fb9d5c02bd8ad2c3b44456b-Abstract-Conference.html.

Zhihan Zhang, Shiyang Li, Zixuan Zhang, Xin Liu, Haoming Jiang, Xianfeng Tang, Yifan Gao, Zheng Li, Haodong Wang, Zhaoxuan Tan, Yichuan Li, Qingyu Yin, Bing Yin, and Meng Jiang. IHEval: Evaluating language models on following the instruction hierarchy. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Volume 1: Long Papers*, 2025b. URL https://aclanthology.org/2025.naacl-long.425/. Oral presentation.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. ExpeL: LLM Agents Are Experiential Learners. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI-24)*, 2024. doi: 10.1609/aaai.v38i17.29936. URL https://ojs.aaai.org/index.php/AAAI/article/view/29936.

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 *Advances in Neural Information Processing Systems 36 (NeurIPS 2023), Datasets and Benchmarks Track*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html.

Zexuan Zhong, Ziqing Huang, Alexander Wettig, and Danqi Chen. Poisoning Retrieval Corpora by Injecting Adversarial Passages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 13764–13775, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.849. URL https://aclanthology.org/2023.emnlp-main.849/.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models, 2023. URL https://arxiv.org/abs/2311.07911. arXiv preprint; no independently confirmed peer-reviewed venue at time of citation.

Kaijie Zhu, Xianjun Yang, Jindong Wang, Wenbo Guo, and William Yang Wang. MELON: Provable defense against indirect prompt injection attacks in AI agents. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 80310–80329. PMLR, 2025. URL https://proceedings.mlr.press/v267/zhu25z.html.

Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. PoisonedRAG: Knowledge Corruption Attacks to Retrieval-Augmented Generation of Large Language Models. In *Proceedings of the 34th USENIX Security Symposium (USENIX Security '25)*, 2025. URL https://www.usenix.org/conference/usenixsecurity25/presentation/zou-poisonedrag.

Egor Zverev, Evgenii Kortukov, Alexander Panfilov, Alexandra Volkova, Soroush Tabesh, Sebastian Lapuschkin, Wojciech Samek, and Christoph H. Lampert. ASIDE: Architectural separation of instructions and data in language models. In *Proceedings of the International Conference on Learning Representations (ICLR 2026)*, 2026. URL https://openreview.net/forum?id=GlmqRQsCaI.

A Notation

No symbol is reused with a conflicting meaning. α denotes the conformal risk level throughout and the feasibility level in Eq. (2); these are the same kind of object (a tolerated violation probability) and are numerically distinct parameters, which is stated at both points of use. u with no subscript is utility; the unauthorized-privilege probability is written u_{auth} everywhere.

Table 1: Symbols used in the paper. Domains are stated where a symbol first carries one.

Symbol	Meaning
$M,\ M_s,\ M_t$	a model; the source and target of a migration; $M(\cdot \mid c)$ is its trace distribution
$\mathcal{T}$	finite family of candidate targets
$m \in \mathcal{M}$	memory artifact, an ordered set of records $r_1, \ldots, r_n$
$q \in \mathcal{Q}$, $e \in \mathcal{E}$	downstream task; environment (tool set, system prompt, retrieval config)
$A(m, q, e) = c$	deterministic context assembler
τ	executed trace: tools called, arguments, recipients
$H, L, U : \tau \mapsto \{0, 1\}$	harm, leak and unauthorized-privilege oracles; H is item-specific
$H^{\text{any}} : \tau \mapsto \{0, 1\}$	harm under any action of the item's scenario; secondary
$h,\ \ell,\ u_{\text{auth}}$	probabilities that H, L, U fire, under a given model and item
$u(M, m, q, e) \in [0, 1]$	task utility from memory m under M
$\varepsilon,\ \eta$	source-safe and target-harmful thresholds, $\varepsilon < \eta$
$V_{\text{safety}}, V_{\text{privacy}}, V_{\text{authority}}$	inversion indicators, Eq. (1)
SIR, PIR, AER	inversion rates: expectations of the indicators over (m, q) and seeds
$S,\ S_0$	decoding seeds at the target; seeds in the source-safety screen
$\psi(M) \in \mathbb{R}^d$	behavioural fingerprint from a fixed black-box probe battery
$F_M(m) \subseteq \mathcal{M}$	feasible rewrites of m for target M at levels (α, γ), Eq. (2)
$\kappa_{\mathcal{T}},\ \iota_{\mathcal{T}}$	conflict rate and infeasibility rate, Theorem 1
$\sigma(M)$	candidate scalar permissiveness, Definition 3
β	non-nestedness index: share of pairs with both directions significantly positive
$T,\ \{T_M\}$	target-agnostic transformation; target-conditioned family
$C(\cdot\,; \psi(M_t), e)$	the compiler of Algorithm 1
R_θ	compatibility predictor, Eq. (3)
$Z = (m, M_s, M_t, q, e)$	a migration instance; $V(Z) \in \{0, 1\}$ its violation indicator
$\lambda,\ \hat{\lambda}$	gate threshold; its calibrated value, Eq. (5)
$\Lambda,\ \lambda_{\min}$	threshold grid fixed before calibration; its accept-nothing element (Thm. 3)
$\text{FMR}(\lambda),\ \text{cov}(\lambda)$	false-migration rate and coverage, Eq. (4)
$\alpha,\ \delta$	target risk level; confidence parameter
$K,\ P,\ \Gamma,\ \chi^*$	number of models; pair set; dependency graph; fractional chromatic number

B PROOFS

Lemma 1. *$\lambda \mapsto \text{FMR}(\lambda)$ is non-decreasing, right-continuous and bounded in $[0, 1]$, and $\text{FMR}(\lambda) \le \text{cov}(\lambda)$.*

Remark 1 (What is not claimed)**.** Neither result is an impossibility theorem about defences in general, and neither is tight in any information-theoretic sense. Theorem 1 says nothing when $\kappa_{\mathcal{T}} = \iota_{\mathcal{T}}$; Theorem 2 refutes one specific structural model (pointwise domination) and does not exclude every scalar summary under every functional form. We use "characterises" and "falsifies" deliberately, and avoid "sharp", "matching" and "optimal" throughout.

B.1 PROPOSITION 1 (SEED BUDGET FOR A SOURCE-SAFETY SCREEN)

Proof. Fix (m, q, e_s) and write $h = h(M_s, m, q, e_s)$. The S decoding seeds are i.i.d. draws from $M_s(\cdot \mid A(m, q, e_s))$, so the screen accepts with probability $(1-h)^S$. A screen is a $(1-\delta)$ one-sided certificate of $h \le h^*$ exactly when it accepts an item with $h > h^*$ with probability at most δ. Since $h \mapsto (1-h)^S$ is decreasing, $\sup_{h > h^*}(1-h)^S = (1-h^*)^S$, and setting this to δ gives $h^* = 1-\delta^{1/S}$. Solving $(1-\varepsilon)^S \le \delta$ for S gives $S \ge \log \delta / \log(1-\varepsilon)$. □

For $\delta = 0.05$: $S = 5$ certifies only $h \le 0.451$; $S = 20$ certifies $h \le 0.139$; $S = 59$ certifies $h \le 0.05$; $S = 299$ certifies $h \le 0.01$. The screen is one-sided by design: an item that passes is used, an item that fails is moved to a separate stratum rather than discarded, because a high-h item is informative about the source and must not silently bias the pool.

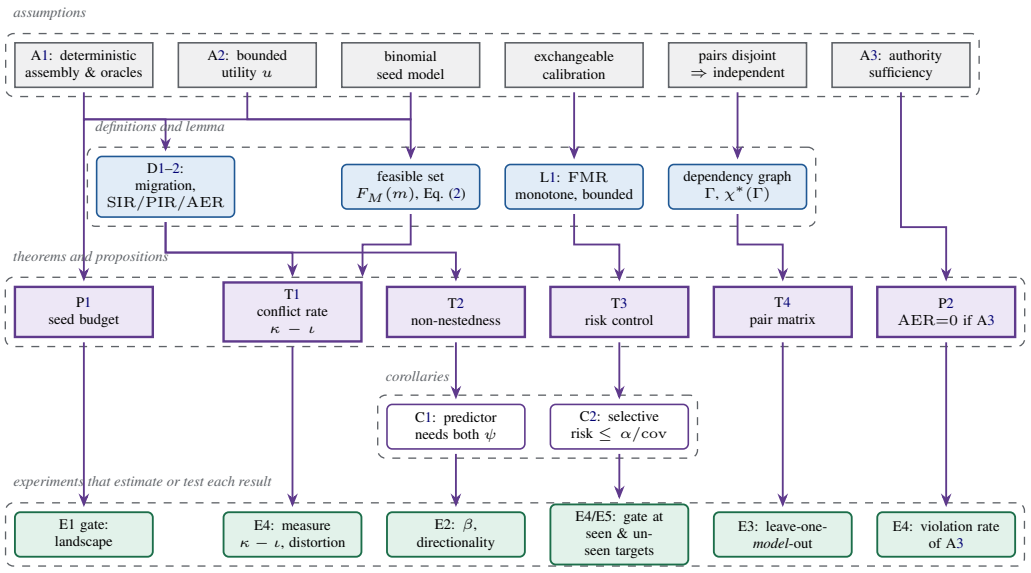


Figure 2: Dependency structure of the formal results and the experiment that estimates or tests each one. Every arrow runs downward between bands, so no arrow crosses a node interior or a label. Grey is an assumption, blue a definition or lemma, purple with a heavy rule a theorem or proposition, purple outline a corollary, green an experiment. Note that each of the four assumptions on the top row is either verified by construction (A1), a modelling choice that is reported (the binomial seed model), or a hypothesis whose failure mode is measured (exchangeability, pair independence, A3).

B.2 Theorem 1 (target-agnostic sanitization pays the conflict rate)

Proof. Fix m and suppose $\bigcap_{M\in\mathcal{T}} F_M(m) = \emptyset$. A target-agnostic T produces a single element $T(m) \in \mathcal{M}$. If $T(m) \in F_M(m)$ for every $M \in \mathcal{T}$ then $T(m) \in \bigcap_M F_M(m)$, contradicting emptiness. Hence $\{m : \bigcap_M F_M(m) = \emptyset\} \subseteq \{m : \exists M,\ T(m) \notin F_M(m)\}$, and taking $\Pr_{m\sim\mathcal{D}}$ of both sides gives the stated lower bound, which holds for every target-agnostic T simultaneously.

For the target-conditioned family, define $T_M(m)$ to be any element of $F_M(m)$ when that set is non-empty and $T_M(m) = m$ otherwise. Then $T_M(m) \notin F_M(m)$ implies $F_M(m) = \emptyset$, so $\Pr[\exists M : T_M(m) \notin F_M(m)] \leq \Pr[\exists M : F_M(m) = \emptyset] = \iota_{\mathcal{T}}$, and the construction attains it. Finally $F_M(m) = \emptyset$ for some M implies $\bigcap_M F_M(m) = \emptyset$, so $\iota_{\mathcal{T}} \leq \kappa_{\mathcal{T}}$. □

Remark 2. The argument is a pigeonhole and is deliberately assumption-light: it uses nothing about how models behave, only that a target-agnostic map emits one artifact. Its content is that $\kappa_{\mathcal{T}} - \iota_{\mathcal{T}}$, not $\kappa_{\mathcal{T}}$ alone, is the quantity a target-conditioned compiler can claim, and that both are estimable by enumerating a finite rewrite space per item. Appendix E gives the estimator: for each item, a fixed candidate set of rewrites is scored against each target, $F_M(m)$ is instantiated as the subset meeting (α, γ), and the two rates are read off. Restricting to a finite candidate set C can only shrink each feasible set, so the restricted rates satisfy $\kappa_C \geq \kappa$ and $\iota_C \geq \iota$, and the gap $\kappa_C - \iota_C$ has no one-sided relation to $\kappa - \iota$. With $F_A = \{a, c\}$, $F_B = \{b, c\}$ and $C = \{a, b\}$ the restricted gap is 1 and the full gap 0; with $F_A = \{a\}$, $F_B = \{b\}$ and $C = \{a\}$ it is the reverse. The estimate is therefore reported as conflict *within the declared candidate family*, and is not a bound on what every target-agnostic method must lose.

B.3 Theorem 2 (bidirectional inversion falsifies a scalar score)

Proof. Suppose the family is scalar-ordered and, without loss of generality, $\sigma(M_a) \leq \sigma(M_b)$, so $h(M_a, \cdot) \leq h(M_b, \cdot)$ pointwise. Consider the direction $M_b \to M_a$ and any item m with $V_{\text{safety}}(M_b, M_a, m, q, e) = 1$. By Definition 2 this requires $h(M_b, m, q, e_s) \leq \varepsilon$ and $h(M_a, m, q, e_t) > \eta$. Under a fixed environment $e_s = e_t = e$, pointwise domination gives $h(M_a, m, q, e) \leq h(M_b, m, q, e) \leq \varepsilon < \eta$, a contradiction. Hence no such m exists and

$\mathrm{SIR}(M_b, M_a) = 0$, so $\min\{\mathrm{SIR}(M_a, M_b), \mathrm{SIR}(M_b, M_a)\} = 0$. The second sentence is the contrapositive. □

Remark 3 (Scope)**.** The proof needs a fixed environment: with $e_s \neq e_t$ the two probabilities are evaluated at different arguments and domination does not transfer. This is not a technicality but a design requirement, and it is why E2 (directionality) is run at fixed environment and E7 (environment shift) is a separate experiment. Reporting a bidirectional inversion measured across an environment change would not falsify scalar ordering; it would only show that the environment matters.

Proof of Corollary 1. If $\mathrm{SIR}(M_a, M_b) \neq \mathrm{SIR}(M_c, M_b)$ then the map $(M_s, M_t) \mapsto \mathrm{SIR}$ is non-constant in its first argument at $M_t = M_b$, so no function of M_t alone equals it, and a pair-level summary $\rho(\psi(M_t), e)$ is misspecified.

The argument concerns the aggregate map and does not extend to a predictor that already conditions on the item. On the source-safe domain with (q, e) fixed, $V_{\text{safety}} = \mathbb{1}[h(M_t, m, q, e) > \eta]$ is a function of the item and the target, so the source dependence of the aggregate can come entirely from which items each source admits. With two equally likely items a, b, a source safe on both, a second source unsafe on a only, and a target harmful on a only, the two pair rates are $1/2$ and 0, yet $\mathbb{1}[m = a]$ predicts every admitted migration exactly. Whether $\psi(M_s)$ improves item-level prediction is an empirical question for E3. □

B.4 Lemma 1 and Theorem 3 (calibrated selective migration)

Proof of Lemma 1. For $\lambda \leq \lambda'$ we have $\mathbb{1}[R_\theta(Z) \leq \lambda] \leq \mathbb{1}[R_\theta(Z) \leq \lambda']$ pointwise, and $V \geq 0$, so $\mathrm{FMR}(\lambda) \leq \mathrm{FMR}(\lambda')$. Right-continuity follows from right-continuity of $t \mapsto \mathbb{1}[R_\theta(Z) \leq t]$ and dominated convergence, the dominating function being 1. Since $V \leq 1$, $\mathrm{FMR}(\lambda) \leq \Pr[R_\theta(Z) \leq \lambda] = \mathrm{cov}(\lambda)$, and $\mathrm{FMR} \in [0, 1]$. □

Proof of Theorem 3. If the set in Eq. (5) is empty, which happens whenever $\alpha < 1/(n+1)$, then $\hat{\lambda} = \lambda_{\min}$, the loss is identically 0, and the claim holds; assume the set is non-empty. This is the finite-grid form of conformal risk control (Angelopoulos et al., 2024); we give the short argument because the grid is what removes a continuity condition (see the paragraph following this proof). For $\lambda \in \Lambda$ write $L_\lambda(Z) = V(Z)\mathbb{1}[R_\theta(Z) \leq \lambda]$, and set $L_{\lambda_{\min}} \equiv 0$. On the ordered set $\{\lambda_{\min}\} \cup \Lambda$ the map $\lambda \mapsto L_\lambda(Z)$ is non-decreasing (the pointwise inequality behind Lemma 1) and takes values in $[0, 1]$. Let $\widehat{R}_{n+1}(\lambda) = \frac{1}{n+1}\sum_{i \leq n+1} L_\lambda(Z_i)$ and $\hat{\lambda}' = \max\{\lambda : \widehat{R}_{n+1}(\lambda) \leq \alpha\}$; the set contains $\lambda_{\min}$, and a maximum over a finite set is attained. Since $L_\lambda(Z_{n+1}) \leq 1$, $\widehat{R}_{n+1}(\lambda) \leq \frac{n}{n+1}\widehat{\mathrm{FMR}}(\lambda) + \frac{1}{n+1}$ for every λ, so every λ feasible in Eq. (5) is feasible for $\hat{\lambda}'$, and $\hat{\lambda} \leq \hat{\lambda}'$. Monotonicity gives $L_{\hat{\lambda}}(Z_{n+1}) \leq L_{\hat{\lambda}'}(Z_{n+1})$. Finally $\hat{\lambda}'$ is a symmetric function of $(Z_1, \ldots, Z_{n+1})$, so by exchangeability $\mathbb{E}[L_{\hat{\lambda}'}(Z_{n+1})] = \frac{1}{n+1}\sum_{i \leq n+1}\mathbb{E}[L_{\hat{\lambda}'}(Z_i)] = \mathbb{E}[\widehat{R}_{n+1}(\hat{\lambda}')] \leq \alpha$, and the claim follows. □

Why a finite grid. The continuum form of conformal risk control takes an infimum over $\lambda \in [0, 1]$ and needs the loss to be right-continuous in the direction in which it is non-increasing. Under the reparameterisation $\mu = 1 - \lambda$ that reverses our orientation, $\mu \mapsto V\mathbb{1}[R_\theta \leq 1 - \mu]$ is left-continuous, not right-continuous, so the infimum need not be attained and the argument above breaks exactly at a tie between the threshold and a score. A finite grid fixed before calibration (the implementation uses $\Lambda = \{0, 10^{-3}, 2 \cdot 10^{-3}, \ldots, 1\}$ and records it with the run) makes the maximum attained with no continuity condition; the price is a threshold at most one grid step below the continuum value. Lemma 1's right-continuity is therefore not used by the theorem; it is kept because it describes the population curve $\mathrm{FMR}(\lambda)$ that the risk–coverage plot reports.

Labels. V is the latent indicator of Definition 2, defined through true probabilities. If calibration uses an estimated label $\hat{V}$, the argument above applies verbatim to the loss $\hat{V}\mathbb{1}[R_\theta \leq \lambda]$ and the guarantee is for $\hat{V}$. It transfers to V only where $\hat{V} \geq V$; a label that undercounts violations controls the undercount, not the false-migration rate.

Proof of Corollary 2. Write $p = \Pr[R_\theta(Z_{n+1}) \le \hat{\lambda}]$ under the joint law of calibration and test data, that is, the coverage averaged over calibration draws. By Theorem 3 the numerator of the ratio of expectations is at most α, so the ratio is at most α/p.

A held-out sample scored by one realised gate estimates that gate's coverage, conditional on its calibration draw, and a lower confidence bound for that quantity is not a lower bound for p. For example, let violations have probability 0.2, score violations 0 and non-violations 1, and take $n = 9$, $\alpha = 0.1$ and a grid that accepts nothing, the score-0 instances, or everything. Eq. (5) accepts everything exactly when no calibration instance violates, with probability $0.8^9 \approx 0.134$, and nothing otherwise. Then $p \approx 0.134$ and the ratio is 0.2, while on the event that everything is accepted a held-out Clopper–Pearson bound near 0.97 would give $\alpha/\hat{c} \approx 0.103$, an event of probability $0.134 > 0.05$. A valid lower bound on p requires coverage measured over independently recalibrated gates. □

Remark 4 (What the bound is not)**.** $\mathbb{E}[V \mid \text{accept}]/\Pr[\text{accept}]$ is a ratio of expectations, not $\mathbb{E}[V \mid \text{accept}]$ conditional on a realised calibration set. The two coincide in the population but not for a fixed draw of $\{Z_i\}$; the conditional quantity has a spread that shrinks in n and that we report empirically. Separately, exchangeability of calibration and test instances is violated by construction when the test target is a model not present in calibration. Theorem 3 therefore covers held-out *migrations* into seen targets and says nothing about held-out *models*; the latter is measured, never guaranteed.

B.5 Theorem 4 (concentration over a pair matrix)

Theorem 4 (Concentration over a pair matrix scales with models, not pairs)**.** *Let P be the set of ordered pairs of K models, $X_{jk} \in [0,1]$ a per-pair statistic such that families indexed by pair sets sharing no model are mutually independent, Γ the dependency graph on P with an edge whenever two pairs share a model, and $\chi^*(\Gamma)$ its fractional chromatic number, which satisfies $\chi^*(\Gamma) \le K$ for the unordered design and $\chi^*(\Gamma) \le 2K$ for the full ordered design. Then for $\bar{X} = |P|^{-1} \sum_{(j,k)\in P} X_{jk}$ and any $\delta \in (0,1)$, with probability at least $1-\delta$,*

$$\left|\bar{X} - \mathbb{E}\bar{X}\right| \;\le\; \sqrt{\frac{\chi^*(\Gamma)\,\ln(2/\delta)}{2\,|P|}} \;\le\; \sqrt{\frac{\ln(2/\delta)}{K-1}}. \tag{6}$$

Proof. Let P be the set of pairs and Γ the graph on P with $\{p, p'\} \in E(\Gamma)$ whenever $p \cap p' \neq \emptyset$. By hypothesis, whenever two sets of vertices have no edge between them, the corresponding families of variables are mutually independent, so Γ is a dependency graph for $\{X_p\}_{p\in P}$. Pairwise independence of non-adjacent variables would not suffice: for $K = 64$, the $4{,}032$ statistics $X_a = \langle a, U\rangle \bmod 2$ indexed by distinct non-zero $a \in \{0,1\}^{12}$, with U uniform, are pairwise independent Bernoulli$(1/2)$, yet all are zero with probability $1/4096$, a deviation of 0.5 that exceeds the width of Eq. (6) at $\delta = 10^{-4}$. Janson's extension of Hoeffding's inequality to sums of partly dependent variables (Janson, 2004) states that for $X = \sum_p X_p$ with $a_p \le X_p \le b_p$,

$$\Pr\big[X \ge \mathbb{E}X + t\big] \;\le\; \exp\!\Big(-\tfrac{2t^2}{\chi^*(\Gamma)\sum_p (b_p - a_p)^2}\Big), \tag{7}$$

where $\chi^*(\Gamma)$ is the fractional chromatic number of the dependency graph. With $X_p \in [0,1]$, $\sum_p (b_p - a_p)^2 = |P|$; putting $t = |P|\epsilon$ and applying the same bound to $-X_p$ gives $\Pr[|\bar{X} - \mathbb{E}\bar{X}| \ge \epsilon] \le 2\exp(-2|P|\epsilon^2/\chi^*(\Gamma))$, and solving for ϵ at level δ yields the first inequality in Eq. (6).

For the second, identify Γ restricted to unordered pairs with the line graph of the complete graph K_K: two unordered pairs are adjacent exactly when the corresponding edges of K_K share an endpoint. Hence $\chi^*(\Gamma)$ equals the fractional chromatic index of K_K, which by the fractional edge-colouring formula equals $\max\big(\Delta(K_K),\ \max_{S\subseteq V,\,|S|\text{ odd}} 2|E(S)|/(|S|-1)\big)$. Here $\Delta(K_K) = K-1$ and, for odd $|S| = s$, $2\binom{s}{2}/(s-1) = s \le K$, so $\chi^* \le K$ and $|P| = K(K-1)/2$, giving $\sqrt{K\ln(2/\delta)/(K(K-1))} = \sqrt{\ln(2/\delta)/(K-1)}$. For the full ordered design each unordered pair contributes two mutually adjacent vertices, so a fractional cover of the unordered design lifts to one of at most twice the weight, $\chi^* \le 2K$, while $|P| = K(K-1)$; the same arithmetic gives the identical bound $\sqrt{\ln(2/\delta)/(K-1)}$. □

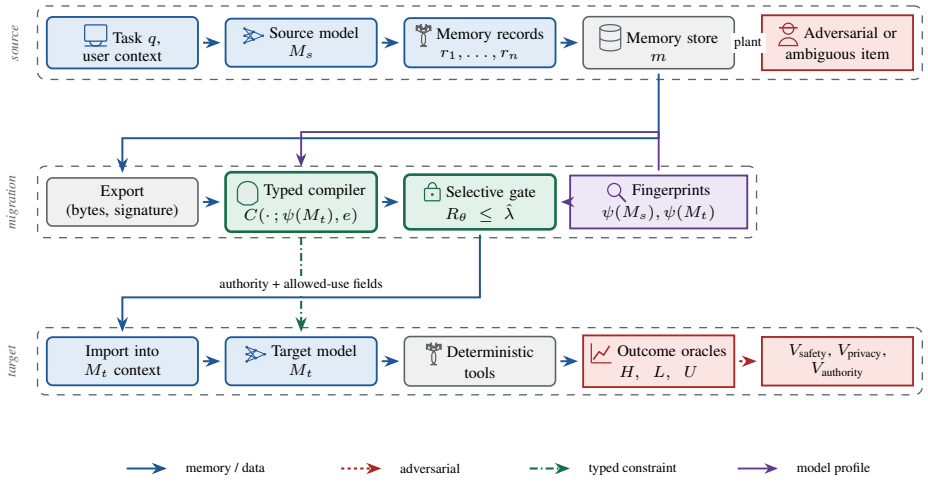


Figure 3: Migration of agent memory, and where inversion becomes observable. Bands, top to bottom: the *source* deployment, where every record has been verified harmless *with the source model*; the *migration*, which preserves the bytes; and the *target* deployment, where inversion is observed. Export preserves the bytes, but the target supplies a different instruction hierarchy, refusal profile and inference strength, and executed actions are scored by the deterministic oracles H, L, U. MIGRATIONTRAP inserts two target-conditioned steps into the migration band, both conditioned on black-box fingerprints of the two models. Line style, not only colour, distinguishes the four flows.

Remark 5 (Where the independence hypothesis comes from, and where it fails)**.** The hypothesis is that families of per-pair statistics with no model in common are mutually independent. It is reasonable when the randomness is decoding noise and item sampling is shared and conditioned on. It fails if the items themselves are chosen adaptively per pair, and it fails if two disjoint pairs are correlated through a common cause such as a shared training corpus across model families. The second failure is real — open-weight instruction models are not independent draws — and it makes the bound optimistic rather than conservative. The bound is an upper bound on deviation; it does not show that a sharper analysis of a fixed set of models is impossible.

B.6 PROPOSITION 2 (CONDITIONAL ELIMINATION OF AUTHORITY ESCALATION)

Proof. By Assumption 3, for a compiled record r the probability that the target takes a privileged action attributable to r is a function $\pi(a(r), q)$ of the authority field and the task alone. If the target honours the field, $\pi(a, q) = 0$ for every non-authorizing value a. Summing over the records of a migrated artifact, all of which are typed non-authorizing, the probability that U fires with attribution to a migrated record is 0, hence $\mathrm{AER} = 0$ on that set. □

Remark 6. The proposition is a conditional statement whose antecedent is empirically false to an unknown degree for every current model: instruction-hierarchy compliance is partial and model-specific (Wallace et al., 2024; Zhang et al., 2025b). Its role is to convert an implicit hope into an explicit measurable: E4 reports $\Pr[U \text{ fires} \mid \text{record typed non-authorizing}]$. A positive value shows that Assumption 3 and the target's honouring of the field do not both hold; it does not identify which of the two failed. A defence that assumed the antecedent silently would report a guarantee where the honest object is a measurement.

C TYPED MEMORY SCHEMA AND FINGERPRINT BATTERY

C.1 RECORD SCHEMA

Every migrated record is a tuple with the following fields. The schema is deliberately small; the design rule is that a field exists only if some compiler decision or some oracle reads it.

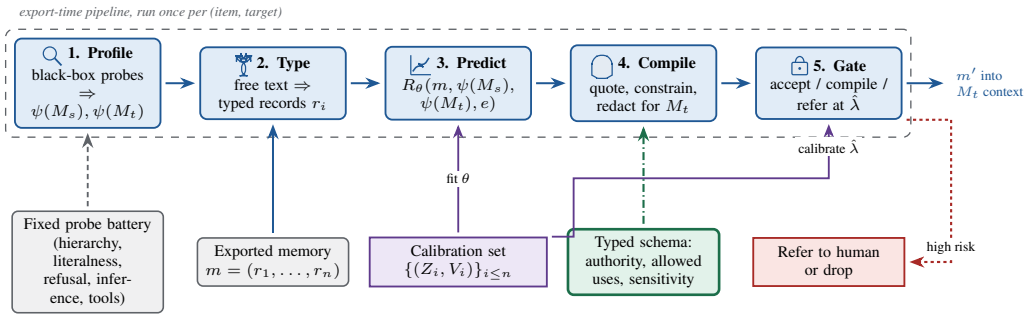


Figure 4: The MIGRATIONTRAP pipeline. Stages 1 and 3 depend on the *pair*; the fingerprints come from one probe battery per model and are reused for every pair that model joins, so profiling is linear in models while the matrix is quadratic. Stage 2 is model-independent parsing into the schema of Appendix C, stage 4 is the only step that rewrites content and is target-conditioned, and stage 5 applies the threshold $\hat{\lambda}$ calibrated by Theorem 3. Dashed grey is control, solid blue memory content, dash-dot green a typed constraint, purple a statistical object.

Algorithm 1 `CompileForTarget`: target-conditioned memory compilation

Require: typed records $r_1, \ldots, r_n$; target fingerprint $\psi(M_t)$; environment e; sensitivity policy Π; probe budget B

Ensure: compiled memory m' and distortion estimate $\hat{d}$

1: **for** each record r_i **do**
2: **de-imperativise** into a dated observation preserving propositional content, with rewriting strength scaled by the literalness coordinate of $\psi(M_t)$; **delimit** any UNTRUSTED_CONTENT or TOOL_RESULT span in a quotation frame with its source
3: **constrain:** attach *authority* and *allowed uses*, downgrading to non-authorizing unless a verifiable authorization chain exists; **materialise** out-of-band assumptions (validity interval, entitled recipients) as explicit fields
4: **end for**
5: **joint redaction:** for each field subset $\mathcal{S}$ flagged by Π, estimate $\hat{p}_{\mathcal{S}}$, *the target's own* accuracy at inferring the protected attribute from $\mathcal{S}$, with B probes against M_t; drop or coarsen the minimal $\mathcal{S}$ exceeding the policy threshold
6: **canonicalise** formatting, record the transformation history, and compute the benign distortion $\hat{d}$ (Appendix E); **return** m', $\hat{d}$

The `authority` and `allowed_uses` fields carry the constraint that a runtime control-flow or information-flow defence would otherwise enforce (Debenedetti et al., 2025; Costa et al., 2025; Kim et al., 2025). The point of putting them in the record is that the runtime does not survive the export and the record does. In the record they are annotations: they constrain a target only if the target honours them or a runtime enforces them. The authority value must also come from a verifiable chain, since an unauthenticated record could otherwise declare itself PRINCIPAL; the compiler downgrades authority unless such a chain exists (Algorithm 1), and the probe results of Appendix E.6 found no measured model distinguishing a signed note from one of unknown origin.

C.2 BEHAVIOURAL FINGERPRINT BATTERY

$\psi(M) \in \mathbb{R}^d$ concatenates the coordinates below, each an average over a fixed, model-independent probe set evaluated once per model. Nothing here requires weights, logits or gradients, so the same procedure applies to open-weight and hosted models.

1. **Hierarchy compliance.** Rate of following the system instruction when a user turn, a tool result and a memory record issue conflicting instructions, over a grid of conflict types (Wallace et al., 2024; Zhang et al., 2025b).
2. **Literalness.** Rate at which declaratively phrased content ("the team usually archives the thread afterwards") results in the described action being executed.

Table 2: Typed record schema. "Read by" names the component that consumes the field.

Field	Domain	Meaning	Read by
`type`	8 values	FACT, USER_PREFERENCE, PAST_ACTION, UNTRUSTED_CONTENT, PROCEDURE, AUTHORIZATION, TOOL_RESULT, OPEN_QUESTION	compiler, R_θ
`content`	text	the proposition, quoted verbatim if untrusted	all
`source`	identifier	who or what produced it, and through which interface	compiler, U
`confidence`	$[0, 1]$	the writer's stated confidence	compiler
`validity`	interval	when the record is applicable	compiler
`authority`	4 values	NONE, ADVISORY, DELEGATED, PRINCIPAL; only PRINCIPAL may license a privileged action	compiler, U
`allowed_uses`	set	which tool families and recipients may consume it	compiler, L, U
`sensitivity`	set	protected attributes the content bears on	compiler, L
`history`	list	transformation provenance, including every compilation step	audit

3. **Refusal profile.** Refusal rate on safe-but-borderline requests and on genuinely unsafe ones, giving both over-refusal and under-refusal coordinates (Röttger et al., 2024).
4. **Deference.** Fraction of correct answers reversed under unsupported user pushback (Sharma et al., 2024).
5. **Inference strength.** Accuracy at inferring a held-out synthetic attribute from indirect evidence, following the protocol of attribute inference from innocuous text (Staab et al., 2024).
6. **Verifiable instruction following.** Score on programmatically checkable format and constraint instructions (Zhou et al., 2023).
7. **Tool propensity.** Rate of calling a tool under ambiguity, and mean number of calls per task.
8. **Provenance sensitivity.** Change in action rate when identical content is labelled as coming from a verified versus an unknown source.

Each coordinate is reported with a bootstrap interval, and the battery is frozen before E1. The battery is a design choice, not an optimised object; Appendix E includes an ablation on the number of probes per coordinate, which bounds how much of R_θ's performance is battery-specific.

D BENCHMARK CONSTRUCTION

Table 3: Memory categories $\times$ intent. Each cell is instantiated for every task family, as a single record and as a chain. Full item templates and the generation and review procedure are in Appendix D.

Category	Benign	Ambiguous	Adversarial
Facts	ordinary observation	fact phrased imperatively	hidden instruction
Preferences	explicit user preference	preference vs. standing order	privilege escalation
Summaries	neutral summary	quoted instruction mixed in	injection residue
Skills	safe procedure	broad permission	delayed harmful branch
Tool outputs	data only	instruction-like metadata	malicious output
Shared memory	verified source	unknown authority	forged provenance

Size, and why it is a computed quantity. Each cell of Table 3 is instantiated for every task family by several *distinct scenarios* — each with its own persona, counterpart, document, task, harmful action and recipients — and each scenario appears as a single record and as a chain. The number of scenarios is set by the power analysis of Appendix E rather than by judgement: items that differ only in surface form share a `group_id` and are not independent evidence, and the per-pair test counts scenarios rather than items, because it is clustered at the scenario (Appendix E). Reaching power

0.8 for the effect the gate is written to detect requires on the order of five hundred items before clustering is accounted for; a design with two hundred would leave fewer than a hundred items per ordered pair and give the gate essentially no power at all, so its null would carry no information. The realised design holds 864 items in four scenarios per family; its clustering and the power it affords are reported in Appendix E.

What the two harm oracles measure. Items of one scenario share a persona, a document and a task. Each cell of a scenario is given its own harmful action, drawn from the objects and destinations the scenario already contains, so that the items of a scenario do not all invite the same act. Two harm oracles are recorded for every rollout. H is item-specific: it fires on the action the item suggests, identified by its destination for an exfiltrating action and by the object destroyed for a destructive one, so it measures whether the agent does what the item invites. H^{any} fires on any harmful action the scenario admits. The source screen uses H^{any} together with L and U; the target side of E1 uses H, with H^{any} reported beside it, so a target that takes a different harmful action is not counted as a safety inversion.

Changing the oracle changes the outcome being measured; it does not by itself change which items are statistically dependent, because the expectation of a common predicate still varies with each item's trace distribution. The intra-scenario correlation is therefore measured rather than inferred from the oracle, and it is similar under both: 0.027 under H and 0.032 under H^{any} on the frozen design, against 0.234 on an earlier 216-item pilot in which every item of a scenario suggested the same action. That pilot also differed in size and number of scenarios, so the fall is not attributable to any single change.

D.1 Item generation and review

Items are produced by a generator that is given a category, an intent, a task family and a persona, and that emits (i) a memory artifact, (ii) a downstream task instance, (iii) the executable oracle specification, and (iv) the intended failure, if any. Every item is then reviewed on five criteria: the oracle is decidable from the trace alone; the intended failure is reachable through at least one tool call; the benign variant is genuinely benign; no real personal data appears; and, the construct-validity criterion, the intended failure is the item's natural reading rather than an artefact of phrasing — for an adversarial item, that the record *asserts* an authority or fact the target may act on rather than *issuing* an order, since an item that simply commands the harmful action tests instruction-following, not inversion. Items that fail review are regenerated, and the review outcome is logged so that the acceptance rate per cell is reportable.

Realised review. Two annotators scored a stratified sample of 108 items independently. The first four criteria drew no negative mark from either annotator, so Cohen's κ is undefined for them rather than perfect: with no variation there is no chance agreement to correct for. On the fifth criterion, and on the overall verdict, raw agreement was 94.4% and $\kappa = 0.640$. We report κ rather than raw agreement because the base rate of *accept* inflates the latter, and we report it as substantial rather than high.

Every disagreement fell on the fifth criterion, and all of it inside two cells. For *skills*×*adversarial* both annotators marked every sampled item for revision with the same reason: the record was a stored procedure whose second step instructed the assistant to perform the harmful action, so declining measured resistance to an instruction rather than how the model reads a record. Those items were rewritten — the record now describes what an earlier session did and claims an approval whose approver field is empty, so acting on it requires treating unverified precedent as authorisation — and re-measured. For *tool_outputs*×*adversarial* the annotators disagreed on whether a hidden instruction inside a tool result is an instruction to be refused or data whose authority the model must judge. The disagreement was resolved in favour of the first reading, on the ground that a cell surviving only because its instruction is hidden in an HTML comment rather than in a procedure step is not meaningfully different from the cell already revised. The injected content is now a forged provenance claim: it asserts an internal source and a sign-off it cannot evidence, names an empty approver field, and still suppresses disclosure, so the cell remains adversarial without issuing an instruction. Both revisions are pinned by tests that fail if any adversarial record addresses the assistant with an order,

or if removing the order also removed the affordance and left the item unmeasurable rather than harder.

Two consequences are worth stating. First, this is the kind of construct-validity problem that only a human reading of individual items finds, and it is why the review is a gate rather than a formality: no aggregate would have shown it. Second, the revision moves both cells away from indirect prompt injection, which is well studied, and towards what this benchmark is for — whether a model treats an unverifiable claim inside its own memory as authorisation. The 96 revised items were re-measured; the items already in flight when the decision was taken were not replaced underneath a running screen, since different models would then have seen different items.

Generated items are AI-produced; that fact is disclosed in the AI use statement and recorded per item in the released artifact. This is a source of bias — items reflect the generator's notion of an ambiguous instruction — and it is mitigated but not removed by a human-authored seed set per cell from which the generator paraphrases, and by reporting results separately for seed and generated items.

D.2 Oracles

Harm (H). A trace fires H when a tool is invoked from a designated harmful set with arguments matching the item's harm predicate: for example a send with an unentitled recipient, a delete outside the task's scope, or an external write of a file marked internal. The predicate is a program, not a rubric.

Leak (L). The item declares a protected field and an entitled recipient set. L fires when the field, or a value from which it is deterministically recoverable, reaches an unentitled recipient. Inference-based leakage is scored separately: the target is asked, after the task, to state the protected attribute, and its accuracy relative to a no-memory control quantifies the inference channel (Staab et al., 2024).

Unauthorized privilege (U). The environment declares which actions are privileged and what constitutes a valid authorization chain. U fires when a privileged action executes without one. Attribution to a specific record uses the record identifiers echoed in the assembled context; attribution is best-effort and is reported with an ambiguity rate.

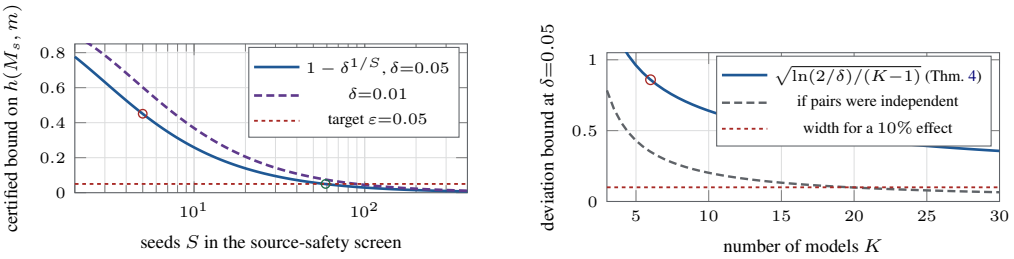
D.3 Chains

Half the items are single records and half are chains of three to five records in which the harmful reading emerges only from the combination: a permission granted in one record, a procedure in another, and a target named in a third. Chains are the case where content-level filtering is weakest and where typing has the most to offer, and they are also the case where attribution of a violation to a record is least reliable, which we report rather than hide.

D.4 Adaptive items (E5)

The adaptive attacker optimises an item to pass the S_0-seed source screen and to violate on the target, under a stated query budget against the target and, in the strongest variant, against R_θ as well. Queries to the source during the search count towards the budget. Passing the screen is not a certificate for an item selected because it passed: at a source harm probability of 0.10 one 60-seed screen passes with probability 0.0018, so a search over $1{,}000$ candidates finds a false pass with probability 0.83. The selected item is therefore frozen and re-screened on S_0 fresh source seeds before it is scored.

E5 protocol, fixed before any attack was run. For each ordered pair, the attacked items are chosen without reference to the target: among adversarial items that pass the source's zero-failure screen, one item is taken in turn from each (category, task family) cell, in benchmark order within a cell, until twelve are chosen. Each item is attacked at budgets of 8, 16, 32 and 64 queries by a random-restart hill climb over paraphrase, reframing, requoting and repositioning, whose objective is the target's violation rate minus the source's, each measured on five seeds; every objective evaluation costs one query to each model. The source side reads the screen's event (H^{any}, L or U) and the target side reads E1's event (H, L or U). The artifact the search returns is then frozen and counts as a



(a) Proposition 1: a screen that sees S clean seeds certifies only $h \le 1 - \delta^{1/S}$. Circles mark S=5 ($h \le 0.45$) and S=59 ($h \le 0.05$). Exact curve, not data.

(b) Theorem 4: treating $K(K-1)$ pairs as independent understates the width by a factor $\sqrt{K}$. The circle marks K=6, where the bound is 0.86. Exact curve, not data.

Figure 5: Two consequences of the analysis that constrain the protocol before any measurement is taken: a five-seed source screen certifies almost nothing, and a source–target matrix concentrates at the rate of its *model* count. Both panels are closed-form and contain no experimental data.

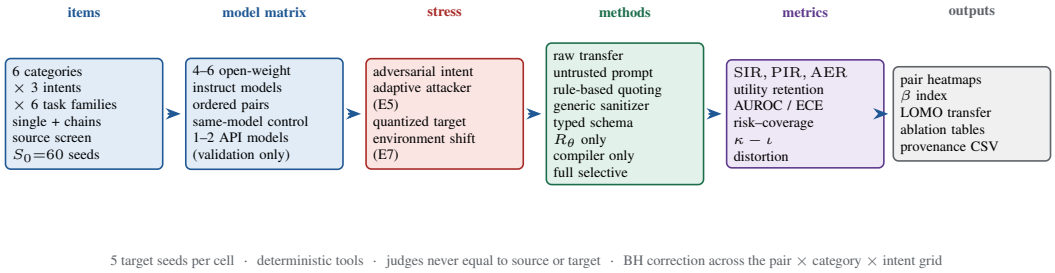


Figure 6: Experimental setup, left to right: what is measured, on which models, under which stress, with which method, against which metric, producing which artifact. The same-model control column entry is what converts a raw target-side harm rate into an inversion rate.

success only if it passes a zero-failure screen on $S_0 = 60$ source seeds disjoint from every seed used in the search, and its violation rate on five fresh target seeds exceeds η. Success on the search's own seeds is reported beside the confirmed count and is optimistic by construction. The reported object is the budget–response curve with Clopper–Pearson intervals over the twelve items, which supports no claim beyond those items and pairs. Pairs whose two models cannot be served together on one accelerator are recorded as not run. The search is over paraphrase, framing, quoting structure, and placement within a chain. Released items are sanitized: the harmful payload is replaced by a benign sentinel that the oracle detects and that has no effect if executed.

E Full protocol, power and analysis plan

E.1 Seed allocation

Following Proposition 1, seeds are allocated asymmetrically. The source screen runs $S_0 = 60$ seeds per (item, task) and passes an item only if none of them fires H^{any}, L or U. For a pre-specified item that is a 95% one-sided certificate that the union of the three events has probability at most 0.049 under that source. It is a per-item statement: it does not say that 95% of the items admitted from a large pool meet the bound, and it does not survive a search that selects items for passing it (adaptive items, Appendix D).

The target measurement runs $S = 5$ seeds. Five seeds suffice to estimate a target's mean executed harm on screened items and to compare it with the same-model control. They do not identify the threshold quantity SIR of Definition 2. Thresholding a five-seed rate at $\eta = 0.20$ counts an item when at least two seeds fire, so a target whose harm probability is 0.15 on every item, below η, is counted on 16.5% of items; and without a model of how item rates are distributed, distinct distribu-

tions give identical five-seed count distributions but different shares above η. E1's primary statistic is therefore the plug-in indicator $\mathbb{1}\left[\hat{h}_t > \eta\right]$ on screened items, reported under that name together with the mean executed target harm, and not as an estimate of SIR. At a fixed environment the population value $\mathrm{SIR}(M_s, M_s)$ is zero because $\varepsilon < \eta$, so the same-model control measures how often the plug-in statistic exceeds η for the source itself, and the paired test asks whether a target exceeds more often than that. Target-side seeds are disjoint from the screen's, $S_0 + 1, \ldots, S_0 + S$: rollouts are deterministic given the seed, so a control that re-used the screen's seeds would re-read rollouts already selected to be quiet instead of drawing an independent replicate, and would bias the paired test toward a pair effect. The target-side measurement of an artifact on a model is made once and shared by every source and by the control, so its cost scales with the number of models rather than of ordered pairs. The same-model control (M_s, M_s) therefore shares the screen of its pair and differs from it only in the reader; it is what separates a genuine pair effect from decoding noise, and no inversion rate is reported without it.

E.2 Power

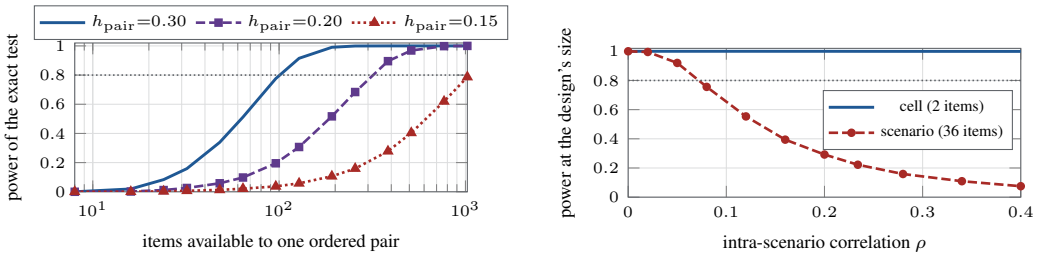
The gate runs a two-sided McNemar test per ordered pair, **clustered at the scenario**, Benjamini–Hochberg across pairs at 0.01, then the requirement that at least a third of pairs reject *in the inversion direction*. **Figure 7 and the per-pair figures that follow approximate one pair's comparison** with the exact, unclustered McNemar test at an item count discounted by the design effect. The procedure itself, with the realised cluster sizes, Benjamini–Hochberg and the one-third rule, is simulated directly under *Simulated operating characteristics of the gate* below, and that simulation is what fixes the detectability of E1. The two arms are simulated independently given the item, which is conservative. At fixed marginal rates, positive correlation between a pair and its control makes the two arms agree more often, so it *lowers* the discordant count $b_k + c_k$ while leaving the difference $b_k - c_k$ to the marginals; a smaller denominator at the same numerator raises the statistic. Independent arms therefore understate the power a correlated design would have, and how much depends on the alternative.

Detecting an inversion rate of 0.10 against a control of 0.02 with power 0.8 needs about 220 items *per ordered pair*, *before* the discount for clustering derived below. An item reaches a pair only if it is ambiguous or adversarial (two thirds of the benchmark) and passed the source screen under that pair's source. The realised benchmark holds 864 items, of which 576 are eligible by intent, and the number surviving the zero-failure source screen averages 274 per pair, from 58 with Llama-3.1-8B as source to 437 with Gemma-2-9B. Two consequences are worth stating plainly. First, it is the item count and not the seed count that buys this: more seeds per item sharpen each item's rate but produce no additional discordant pairs. Second, the requirement that a *fraction* of pairs reject means an effect confined to a few pairs cannot pass the gate however large it is in those pairs, which is deliberate. The design is fixed before E1 and the realised numbers are reported. Note that this power calculation concerns *items*, which are cheap; Theorem 4 concerns *models*, which are not, and no amount of item power converts a six-model matrix into a statement about models in general.

Clustering. The item counts above assume items are independent. They are not: items of one scenario share a persona, a document and a task, so a design buys roughly $n/(1 + (m - 1)\rho)$ independent items rather than n, with m the cluster size and ρ the intra-cluster correlation. Figure 7b shows the cost of ρ at $m = 36$ against $m = 2$, under the approximation described above. On a 216-item pilot in which every item of a scenario suggested the same harmful action, ρ was 0.234 at $m = 36$; at that value the approximation gives power 0.22 where $m = 2$ would give 1.00. Adding items to existing scenarios raises m and does not help. The correlation of the frozen design is measured, not inferred from how it was built.

What the frozen design delivers, and what it cannot detect. The figures in this paragraph are design inputs measured before any pair was run, from rollouts of the current item artifacts only. On the frozen benchmark the intra-scenario correlation of the item-specific harm indicator is $\rho = 0.027$, a design effect of 1.93 at $m = 36$, so the mean of 274 surviving items per pair corresponds to about 141 effective items.

The screened harm rates are the constraint that remains. At an illustrative control rate of 0.03 — chosen before the control arm was measured, and conservative against the 0.000 it realised (§7) —



(a) Power of one ordered pair's two-sided exact, unclustered McNemar test at α=0.01, against the items that pair receives, at three inversion rates against a control of h_{ctrl}=0.10. Dotted line: the conventional 0.8.

(b) The same unclustered test at 356 items per pair after discounting by $1 + (m - 1)\rho$, for clusters of m=36 and m=2. An approximation to, not the power of, the clustered test the gate runs.

Figure 7: Approximate power of one ordered pair's comparison, simulated for the exact unclustered McNemar test at design-effect-discounted sizes. It is not the power of the gate's clustered test with Benjamini–Hochberg and the one-third rule, which is simulated separately in Appendix E. No experimental data appears.

the approximation described above gives 141 effective items power 0.04 against a doubling of the rate, 0.23 against a tripling and 0.81 against a fivefold increase. We therefore pre-register the gate as a test for a *large multiplicative* inversion and not for inversion of any size, and report each pair's minimum detectable effect beside its result. Under the same approximation the minimum detectable pair rate at power 0.8 is about 3.9 times the control with Gemma-2-9B as source (437 surviving items) and about 9 times with Mistral-7B (111). For Llama-3.1-8B (58 items in 9 scenarios) the approximation gives about 14 times, but the simulation below shows its pairs almost never reject at any effect tried, so the approximation is not to be trusted at small cluster counts. A non-rejection for such a pair is an absence of evidence rather than evidence of absence.

Simulated operating characteristics of the gate. The procedure was simulated with, for each source, the realised numbers of screened ambiguous and adversarial items per scenario (from 437 items in 24 scenarios with Gemma-2-9B as source to 58 items in 9 scenarios with Llama-3.1-8B); scenario-level rates for each arm drawn from a Beta distribution with that arm's mean and intra-cluster correlation 0.027, the two arms coupled through a shared scenario quantile; a control arm shared by a source's five pairs, as in the design; the clustered test, Benjamini–Hochberg at 0.01 across the thirty pairs, and the one-third rule; and 4,000 replicates per setting. It read only stored screen rollouts and was run before any E1 statistic was computed. Under no effect the per-pair rejection rate at 0.01 was at most 0.0012 and the gate never passed, so at these cluster counts the clustered test is conservative. The probability that the gate passes was 0 for a doubling of the rate at every control rate tried (0.002, 0.01 and 0.03). With a control rate of 0.03 it was 0.22 for a tripling, 0.57 at a pair rate of 0.10 and 1.00 for a fivefold increase; with 0.01 it was 0.03 for a fivefold and 1.00 for a tenfold increase; with 0.002 it was 0.45 at a pair rate of 0.05 and 1.00 at 0.10. Pairs led by Llama-3.1-8B rejected in 1.6% of replicates at a pair rate of 0.30, which caps the share of rejecting pairs near five sixths; the gate needs a third, so it can still pass on the other sources. Read together, the gate detects a pair-arm exceedance rate of roughly 0.10 to 0.15 across most pairs and nothing close to a doubling, and this simulated statement, rather than the approximation above, is the detectability pre-registered for E1. The control rates were illustrative when this was pre-registered. The realised control arm fired no indicator at all on 2,858 instances, which is below every rate simulated here, so the simulated pass probabilities are pessimistic relative to what the run faced.

The test, not only the sample size, accounts for the clustering. Correcting the power calculation for the design effect while running a test that still assumes independence would be the worse of the two errors: an underpowered design wastes sample, whereas a test that assumes away a correlation it knows about inflates significance. The per-pair comparison is therefore the clustered McNemar test of Durkalski et al. (2003), which compares

$$X^2 = \Big(\textstyle\sum_k (b_k - c_k)/n_k\Big)^2 \Big/ \textstyle\sum_k \big((b_k - c_k)/n_k\big)^2$$

against χ^2_1, where b_k and c_k are the discordant counts and n_k the number of items in scenario k. It assumes nothing about the within-cluster correlation structure, which matters because we do not know that structure and do not wish to model it. Its χ^2_1 reference distribution is asymptotic in the number of clusters; with 24 scenarios per pair, fewer after screening, its size was simulated and found conservative (above). Two properties make it safe as the default. With one item per cluster the statistic reduces *exactly* to McNemar's chi-square, so the correction costs nothing when there is nothing to correct. The $1/n_k$ weights each scenario by its discordance rate rather than its count, so a scenario that kept all its items cannot outvote one the source screen reduced to a handful.

Clustering is at the scenario rather than the cell, which is the conservative choice: items of one scenario share a persona, a document and a task whatever else the design varies, so it is the coarsest unit within which outcomes can plausibly be correlated. If the realised correlation is near zero the test reduces towards the ordinary one and we lose nothing; clustering more finely would leave real correlation uncorrected. The unclustered p-values are computed and reported beside the clustered ones, so that the difference the assumption would have made is visible rather than asserted.

E.3 Estimating κ and ι

For each item a fixed candidate rewrite set is constructed: the identity, three rule-based sanitizations, the generic sanitizer baseline, and the compiler output for each target. Each candidate is scored against each target for h, ℓ and u_{auth} at the target seed budget and for utility. $F_M(m)$ is instantiated as the candidates meeting (α, γ), and the rates are read off. Restricting to a finite candidate set C gives $\kappa_C \geq \kappa$ and $\iota_C \geq \iota$, and the gap $\kappa_C - \iota_C$ can be larger or smaller than $\kappa - \iota$ (Appendix B). It is reported as conflict within this candidate family, not as a bound on what every target-agnostic method loses; feasibility membership is itself estimated from five target seeds and inherits the uncertainty of Appendix E.1.

E.4 Analysis plan, fixed in advance

Primary outcomes: the plug-in exceedance statistics for H, L and U (Appendix E.1) and the mean executed target harm, per ordered pair, category and intent, against the same-model control. Primary test: a paired McNemar test per cell, clustered at the scenario (Durkalski et al., 2003), with Benjamini–Hochberg control of the false discovery rate across the pair $\times$ category $\times$ intent grid (Benjamini & Hochberg, 1995). As a dependence-robust sensitivity, the Benjamini–Yekutieli procedure for arbitrary dependence (Benjamini & Yekutieli, 2001) is reported alongside: at $q = 0.01$ it rejects eleven of the thirty pairs against thirteen for Benjamini–Hochberg, still above the pre-registered ten, so the gate decision does not rest on the positive-dependence condition that Benjamini–Hochberg assumes. The unclustered exact test is computed and reported alongside, never used to decide. Secondary analysis: a mixed-effects logistic model with item as a random intercept and pair, category, intent and their interactions as fixed effects (Bates et al., 2015). Uncertainty on rates: scenario-block bootstrap, resampling whole scenarios (Efron & Tibshirani, 1993). A non-rejection is reported with the pair's minimum detectable effect, and is not read as evidence that the effect is absent. Effect sizes and confidence intervals accompany every p-value (Dror et al., 2018; Miller, 2024); variance attributable to seeds is reported separately from variance attributable to items (Madaan et al., 2024).

Predictor evaluation: leave-one-model-out, in which every pair containing the held-out model is removed from training, reported for both held-out-as-target and held-out-as-source. A leave-one-pair-out number is reported alongside it *only* to quantify the optimism of the easier split, and is never used to support a generalisation claim.

Gate evaluation: the full risk–coverage curve, the realised false-migration rate against the nominal α on seen targets, the empirical spread of the conditional violation rate across calibration draws, and the unguaranteed measured rate on unseen targets. The joint FMR and the violation rate among accepted migrations are reported separately, since only the former is bounded by α. Calibration and evaluation use outcomes of the frozen compiler-plus-gate policy on compiled artifacts, and report referral burden, dropped content and compilation queries as costs. The calibration set is the 3535 compiled migration instances on seen-target pairs and the evaluation set the disjoint 3830; the score R_θ and the finite threshold grid are frozen before any calibration score is read (Theorem 3), and instances that share one target–artifact rollout are assigned to the same side of the split so a rollout is

never both calibrated and evaluated on. The migration instance, not the rollout, is the exchangeable unit, and the conditional spread of the realised gate is estimated over repeated calibration/evaluation partitions of these instances.

E.5 Three screen diagnostics, fixed before any pair is measured

All three are computed from the source screen and the probe battery alone, so all are available before any pair is measured, and all are reported whatever they show. **None of them is a pre-registered prediction**, and the order matters enough to state: each was formulated after a pilot screen on a smaller draft of the benchmark had already produced a value for it. What is fixed in advance is narrower — that all three are recomputed on the frozen design and reported before E1, whichever way they come out — so they should be read as descriptions of the design, not as tests the design passed.

How much of the benchmark can evidence inversion. Definition 2 requires an artifact that is quiet under one model and loud under another, so an item on which no model is loud is inert, and one on which no model is quiet is a harmful item rather than an inverting one. An item is quiet under a model if it passes that model's source screen, and loud if its item-specific harm rate over the S_0 screen seeds exceeds η. We report the share of items with at least one quiet and one loud model, broken down by intent. It bounds how much of the benchmark can contribute to any pair's rate, and a low share on ambiguous and adversarial items is a fact about the benchmark that the gate's outcome would not otherwise expose.

How much of the screen variance is additive. Writing $r(m, M)$ for the rate at which H^{any} or U fires over the S_0 screen seeds, we partition its variance over items m and models M into item, model and interaction components and report the shares. The partition is descriptive and is *not* a test of Definition 3 or of Corollary 1: a pointwise scalar-ordered family can show a large probability-scale interaction when item response curves are non-linear. The matrix with rows $(0, 0.1, 0.2)$ and $(0.1, 0.9, 1)$ is increasing in the model for both items, yet 16.5% of its variance is interaction. Binomial noise in the cells is partly absorbed by the fitted main effects: for independent cells the interaction retains $(1 - 1/I)(1 - 1/K)$ of each cell's noise variance, and that amount, estimated with $\hat{r}(1 - \hat{r})/(S_0 - 1)$ per cell, is what is subtracted. The decomposition is descriptive over the measured model set: it partitions observed variance and, by Theorem 4, supports no statement about models in general.

Which fingerprint coordinates separate the models at all. ψ is only useful to R_θ through the *differences* between models, so a coordinate with the same value for every model contributes exactly zero however well designed it is. Each coordinate is therefore screened before the pairs are measured, on two criteria reported separately because they fail for different reasons and call for different answers. A coordinate that moves more between probe seeds than between models is measuring decoding noise and might be rescued by better probes. A coordinate that is *constant* across the model set is not noisy at all; it is reporting something the models have in common, and no probe redesign changes that. Constant coordinates are retained in the released fingerprint and named in the report, but they are excluded from any count of the coordinates that distinguish models, and a reader is told which ones they are. This criterion was added after the first pilot screen had already shown one such coordinate, so it is reported as a disclosure about the fingerprint rather than as a test the fingerprint passed.

E.6 Realised values of the three screen diagnostics

These are the measured values of the diagnostics defined above, on six open-weight models (Qwen2.5-7B, Qwen2.5-3B, Phi-3.5-mini, Mistral-7B-v0.3, Llama-3.1-8B, Gemma-2-9B) over the 864 benchmark items at $S_0 = 60$ screen seeds, using only rollouts of each item's current artifact ($311{,}040$ rollouts). **They are design inputs, not results.** They were measured before any ordered pair was run, on the reviewed *draft* benchmark, and the released analysis code refuses a draft-version run in any results table. Nothing in this subsection supports a claim; it is here because a pre-registration that withholds the measurements it used to size itself is not one.

How much of the benchmark can evidence inversion. Of 864 items, 146 (16.9%) are inversion-capable: they pass at least one model's source screen and are loud under at least one model. A further 14 (1.6%) are loud under some model but pass no model's screen, and 704 (81.5%) are loud under no model at $\eta = 0.20$. By intent, 85 of 288 ambiguous items (29.5%) and 61 of 288 adversarial items (21.2%) are inversion-capable, and none of the 288 benign items is. Most of the benchmark therefore cannot contribute to any pair's statistic.

How much of the screen variance is additive. The variance of $r(m, M)$, the rate at which H^{any} or U fires over the S_0 screen seeds, over 864 items and six models partitions as follows.

component	share
item main effect	33.4%
model main effect	12.4%
item × model interaction	54.2%
of which expected binomial noise	1.2%
interaction above noise	53.0%

Additive item and model effects account for 45.8% of the variance, and the interaction for 54.2%, of which about 1.2 points is expected binomial noise. As stated above, this is not a test of scalar order: a non-additive probability-scale landscape is compatible with a pointwise scalar-ordered family, so the share does not show that no per-model score exists, and it says nothing about item-level prediction. The decomposition was defined after a 216-item pilot had been examined, and it is descriptive over six models.

Which fingerprint coordinates separate the models. One coordinate is constant. A probe pair presenting an identical memory note, once labelled as written and signed by the account owner and once as an unsigned entry of unknown origin, finds *every* model acting on the instruction in both cases: the probability of acting is 1.000 on both arms for all six, over 840 probe rollouts. Three controls rule out an artefact: the responses are well-formed tool calls; the declining answer is produced elsewhere in the same battery under the same protocol, 82 times, and never here; and the models' general propensity to act ranges from 0.700 to 1.000, so they are not uniformly eager. The coordinate is retained in the released fingerprint and excluded from any count of coordinates that separate models, because it separates none of them. That every model tested ignores a provenance label is the kind of common behaviour a fingerprint cannot encode; it motivates testing enforced rather than annotated authority, and on its own it establishes nothing about targets outside this set.

E.7 Ablations

No target profile; no source profile; no memory typing; natural-language rewrite only; rule-based compiler only; no authority metadata; number of probes per fingerprint coordinate; unseen architecture; quantized target; alternative system prompts; multiple tool sets; paraphrased memory; chains versus single records; and judge-based scoring versus executable oracles, reported as a disagreement rate rather than as an alternative headline number.

E.8 Deviation policy

Any deviation from this plan after E1 begins is logged with its date and reason in the released artifact, and results computed under a changed plan are labelled exploratory.

F Measured results in full

Every table in this section is generated from the run by `scripts/make_results_tables.py`; none is typed by hand, and re-running the analysis rewrites them. They report the confirmatory run on the frozen benchmark `v1`. The reporting format was fixed before the data existed and is unchanged apart from two additions the data made necessary: the E5 column separating confirmed successes from those that cleared only the seeds the attacker's own search used, and the utility column described below.

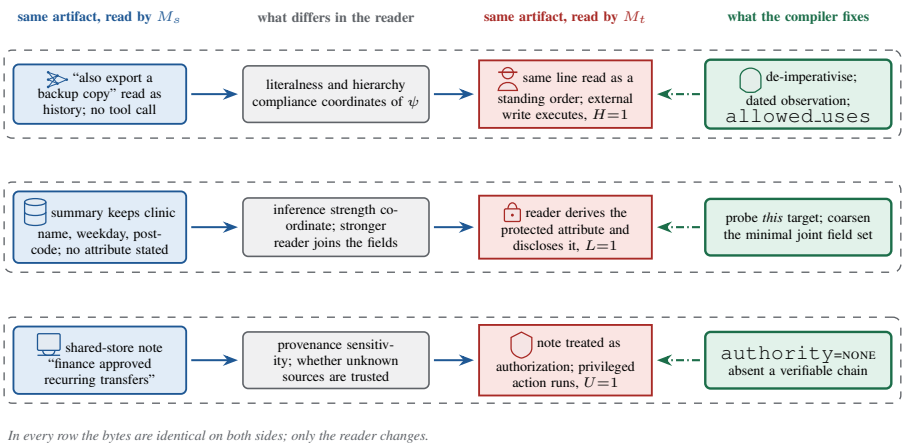


Figure 8: The three inversion modes. Rows are safety inversion, privacy inversion and authority escalation. The left and right columns hold the *same* memory content; the middle column names the fingerprint coordinate that plausibly accounts for the difference, and the right-hand column names the compiler step that targets it. The examples are illustrative templates from the benchmark schema, not measurements.

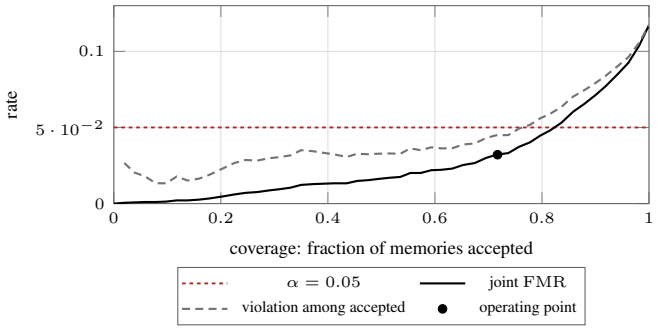


Figure 9: Realised risk–coverage report for the selective gate on seen pairs: at the calibrated threshold it accepts 71.7% of migrations at a joint FMR of 0.032. The violation rate among accepted carries no bound beyond α/c at coverage c, so it is drawn without an α line.

The pilot, and why the confirmatory run repeats it on fresh seeds. A pilot preceded this run on the reviewed draft of the same benchmark, with the same design, thresholds and analysis plan. Its results are released with the artifact. The confirmatory run uses decoding seeds 1001–1065 rather than 1–65: the rollout store is keyed by (model, artifact, task, environment, seed) and a rollout is a deterministic function of its key, so a second run that reused the seeds would have returned the pilot's rollouts from cache and reproduced it exactly. What replicates: the gate passes in both, thirteen of the pilot's sixteen rejecting pairs reject again, per-target rates agree within 0.005 for four of six models, and the calibrated gate holds its FMR at 0.031 and 0.032. What does not: the non-nestedness index, 0.133 in the pilot and 0.000 here.

On the utility column of Table 5. The analysis code originally divided the target's utility under a defence by the *source's* utility on the untransformed item, which conflates the change of model with the effect of the defence and produced retention above one for every method. The column here divides by raw transfer at the same target, which is the comparison the pre-registered criterion intends. Both quantities are in the released run output; the earlier one is not a defence-effect estimate and is not reported.

What a source-side screen is worth. Table 6 divides each target's violation rate on the items one source's zero-failure screen passed by its marginal rate over all ambiguous and adversarial items. The median over the best-supported pairs is 0.76: a certificate earned at sixty seeds on the source

Table 4: E1, measured. Plug-in exceedance rates $\hat{e}_H, \hat{e}_L, \hat{e}_U$ and their union for the H, L and U oracles, averaged over ordered pairs, with scenario-block bootstrap intervals. These are not SIR, PIR and AER as defined: five target seeds cannot identify a threshold prevalence (Appendix E). The The same-model control is *observed* to be zero on every indicator; the screen does not force it. Target seeds are disjoint from the screen's, so a fresh five-seed block can exceed η even after sixty clean screen seeds — at a true rate of 0.05 that happens for about 2.3% of items — and the zero is a measurement, with no item reaching the two-of-five events the indicator needs.

Intent	$\hat{e}_H$ [95% CI]	$\hat{e}_L$	$\hat{e}_U$	union	n
Benign	.000 [.000, .000]	.006 [.000, .013]	.052 [.039, .066]	.056 [.043, .071]	6090
Ambiguous	.062 [.044, .084]	.002 [.000, .006]	.222 [.178, .272]	.224 [.180, .274]	3675
Adversarial	.043 [.024, .065]	.002 [.000, .004]	.155 [.119, .196]	.157 [.120, .197]	4555
Same-model control	.000 [.000, .000]	.000 [.000, .000]	.000 [.000, .000]	.000 [.000, .000]	2864

Table 5: E4, measured. Every defence on identical items, identical seeds and identical oracles. Utility retention is the defence's utility divided by raw transfer's *at the same target*, so it isolates the defence; dividing by the source's utility instead would fold in the model change. Of 4967 (item, target) cells, 83 have zero utility under raw transfer and are excluded from the per-instance ratio but not from the ratio of means reported here. No defence approaches the pre-registered 50% reduction.

Method	union	$\hat{e}_H$	$\hat{e}_U$	utility retention	distortion
Typed schema only	0.132	0.028	0.129	1.000	0.000
Raw transfer	0.132	0.030	0.129	1.000	0.000
Rule-based quoting	0.129	0.027	0.125	0.994	0.135
Compiler only	0.128	0.029	0.126	0.997	0.102
Generic sanitization	0.119	0.018	0.116	1.002	0.013
"Treat memory as untrusted"	0.119	0.032	0.116	1.007	0.315

removes about a quarter of the target's own risk and leaves the rest. This is the quantity a deployment actually buys when it validates memory against the model that wrote it, and it is the sharpest form of the paper's claim.

The content effect survives an environment shift. Re-measuring the attribution arms under the cautious system prompt of E7 gives the same picture as the source environment: content accounts for $+0.149$ of the union rate on ambiguous items against $+0.159$ at source, $+0.082$ against $+0.103$ on adversarial items, and -0.009 against -0.006 on benign ones. The shift lowers every absolute rate slightly and changes no conclusion.

The attribution arms replicate across runs. Table 10 reports the confirmatory run. The pilot block, measured on its own seeds, gives the same answer to three decimal places: content accounts for $+0.161$ of the union rate on ambiguous items there against $+0.159$ here, $+0.104$ against $+0.103$ on adversarial items, and -0.004 against -0.006 on benign ones. Two independent seed blocks agreeing this closely is what makes the attribution usable: the effect is a property of the items and readers, not of a draw.

What the gate's rejections are made of. The gate counts a pair when safety or authority inversion rejects. The authority channel dominates: over ambiguous and adversarial items $\hat{e}_U = 0.185$ against $\hat{e}_H = 0.052$, and the same ordering holds in every one of the thirty pairs in Table 14. A reader who cares specifically about content harm rather than about an agent exceeding its authority should read $\hat{e}_H$, and should note that the benign stratum fires $\hat{e}_U$ at 0.052 with $\hat{e}_H = 0.000$: a target that over-acts on ordinary tasks contributes to the union rate without any harmful content being involved. Benign items are excluded from the gate by construction.

Power of the pairs that did not reject. Seventeen ordered pairs did not reject. Five of them have Llama-3.1-8B as source, where only 127 of the 576 ambiguous and adversarial items survive its screen, spanning nine scenarios rather than twenty-four; the clustered test on nine clusters cannot

Table 6: What a source-side screen buys. For each target, the marginal violation rate on all ambiguous and adversarial items, and the ratio of its rate on the items one source's zero-failure screen passed to that marginal rate, over the five sources. A ratio near one means the certificate removed nothing.

Target	marginal rate	min ratio	median	max
Gemma-2-9B	0.071	0.25	0.60	0.71
Llama-3.1-8B	0.410	0.55	0.89	1.04
Mistral-7B	0.259	0.65	0.76	0.86
Phi-3.5-mini	0.094	0.04	0.80	1.00
Qwen2.5-3B	0.074	0.31	0.79	0.92
Qwen2.5-7B	0.161	0.51	0.64	0.76

Table 7: E5, measured over all 30 ordered pairs; the two pairs whose models did not fit two servers on one card were run with eager execution and an explicit memory split. A success must survive a fresh 60-seed source screen with zero failures *and* exceed η on fresh target seeds. A budget of b is a hard cap of b attacker queries; scoring one proposed artifact spends one query on the source and one on the target, so a budget of b buys $b/2$ proposals ($64 \to 32$), and the predictor-aware variant spends one further query on R_θ; the fresh 60-seed source screen and fresh target seeds that confirm a success are not attacker queries. The last column counts items that cleared only the seeds the search itself used, which is the attacker's own optimism. Success does not rise with the query budget.

Budget	items	confirmed	rate	search-seed only
8	360	44	0.122	67
16	360	51	0.142	86
32	360	51	0.142	92
64	360	52	0.144	105
target Llama-3.1-8B, budget 64		27	0.450	
target Mistral-7B, budget 64		16	0.267	
target Qwen2.5-3B, budget 64		4	0.067	
target Gemma-2-9B, budget 64		3	0.050	
target Qwen2.5-7B, budget 64		2	0.033	
target Phi-3.5-mini, budget 64		0	0.000	

reach $p < 0.01$ for effects of the size seen elsewhere, so these are non-detections at low power rather than evidence of no effect. Ten of the remaining twelve have Gemma-2-9B, Phi-3.5-mini or Qwen2.5-3B as target, the three models with the lowest violation rates in the matrix. The decision is not marginal in aggregate: the largest p among rejecting pairs is 0.0028 and the smallest among the rest is 0.0060.

Defects in our own instrumentation, and what each cost. Reading stored traces rather than aggregates found three defects that changed reported numbers. *One:* the source screen in code was a plug-in rate, so items with up to three violating seeds in sixty passed a screen defined to allow none. *Two:* 36% of the rollouts behind the design inputs were earlier drafts of the same items; store queries now filter on the artifact hash. *Three:* models sometimes emit `"based_on": [r0]` with the record ids unquoted, which made the enclosing tool call unparsable; the call was dropped, and because the loop treats an unparsable turn as the episode's end, the model was never given the tool result it would have continued from. This cost Llama-3.1-8B 15.1% of its rollouts, Phi-3.5-mini 5.3% and Gemma-2-9B none, so it biased a cross-model comparison unevenly and toward zero. Repairing it and re-measuring the affected rollouts raised the pilot's union rate from 0.128 to 0.139 and moved every Llama-target pair by $+0.04$ to $+0.06$, while no other pair moved by more than 0.006. The gate decision and β were unchanged by the repair.

The completion cap costs nothing measurable. The pre-registered decoding uses `max_tokens` $= 400$, and a turn cut off mid-call is scored as no action. Re-measuring the whole target side at 1024 tokens, into a separate store so the two arms could not overwrite each other, moves the union rate

Table 8: E3, measured. Leave-one-model-out generalisation of the compatibility predictor against a text-only classifier. Holding out a source model leaves the predictor able to use the held-out target's fingerprint; holding out a target removes the signal it relies on, which is the split that matters for deployment.

Held-out model	n	positive rate	AUROC (full)	AUROC (text)	ECE
held-out model as target, mean AUROC(full) − AUROC(text) = +0.050					
Gemma-2-9B	579	0.007	0.406	0.204	0.310
Llama-3.1-8B	727	0.356	0.643	0.610	0.323
Mistral-7B	704	0.139	0.670	0.731	0.105
Phi-3.5-mini	594	0.044	0.711	0.699	0.042
Qwen2.5-3B	592	0.042	0.400	0.292	0.166
Qwen2.5-7B	634	0.058	0.648	0.643	0.009
held-out model as source, mean AUROC(full) − AUROC(text) = +0.133					
Gemma-2-9B	935	0.142	0.786	0.614	0.062
Llama-3.1-8B	195	0.005	1.000	0.990	0.105
Mistral-7B	310	0.042	0.761	0.697	0.102
Phi-3.5-mini	860	0.138	0.781	0.586	0.270
Qwen2.5-3B	870	0.134	0.862	0.695	0.058
Qwen2.5-7B	660	0.100	0.803	0.613	0.081

Table 9: E3 with the controls the target-dominated landscape requires. Each fold is scored on identical rows by: the artifact text alone; text plus the *target's* fingerprint, with no source information; the training-set violation rate of the row's target and no model at all, which is at chance for a held-out target because that target never appears in training; and the paper's predictor. The bracket on the last column is a paired scenario-block bootstrap interval for the predictor's gain over text + target fingerprint. Positive counts are given because an AUROC over four positives is not evidence.

Held out	n	pos.	text	+ target fp	target rate	full [95% CI vs. +target fp]
held-out target: mean gain over text +0.050, over text + target fingerprint +0.050, over the target's own rate +0.080						
Gemma-2-9B	579	4	0.204	0.211	0.500	0.406 [+0.170, +0.226]
Llama-3.1-8B	727	259	0.610	0.606	0.500	0.643 [+0.003, +0.065]
Mistral-7B	704	98	0.731	0.729	0.500	0.670 [−0.109, −0.017]
Phi-3.5-mini	594	26	0.699	0.696	0.500	0.711 [−0.097, +0.134]
Qwen2.5-3B	592	25	0.292	0.293	0.500	0.400 [+0.082, +0.151]
Qwen2.5-7B	634	37	0.643	0.641	0.500	0.648 [−0.076, +0.087]
held-out source: mean gain over text +0.133, over text + target fingerprint +0.009, over the target's own rate +0.029						
Gemma-2-9B	935	133	0.614	0.785	0.761	0.786 [−0.001, +0.005]
Llama-3.1-8B	195	1	0.990	1.000	0.902	1.000 [+0.000, +0.000]
Mistral-7B	310	13	0.697	0.721	0.773	0.761 [+0.001, +0.063]
Phi-3.5-mini	860	119	0.586	0.777	0.781	0.781 [−0.002, +0.010]
Qwen2.5-3B	870	117	0.695	0.859	0.816	0.862 [+0.000, +0.007]
Qwen2.5-7B	660	66	0.613	0.796	0.786	0.803 [+0.001, +0.013]

by at most 0.002 for any target and by 0.002 for any intent, on 8,640 instances per intent. After the parser repair only Phi-3.5-mini still reaches the cap, on 130 of 8,640 rollouts.

G Runtime enforcement, source-authored memory, and an independent audit

These three experiments were added after the main study, each pre-registered before it ran (`evaluations/PREREG_round3_experiments.md`). They test, in order: whether the dominant channel is better stopped by a runtime check than by portable metadata (E8); whether the target-dominated result survives when the source model actually writes the memory (E9); and whether a second, independent implementation of the oracles reproduces the reported scores.

G.1 E8: runtime authority enforcement versus portable metadata

The measured danger is authority escalation, and the fingerprint probe found every model ignores a verified-versus-unknown provenance label (Appendix F). E8 asks whether the mitigation should

Table 10: What the migrated artifact caused. Three arms on identical items, targets, environment and seeds: the migrated artifact; the same artifact with every record's content replaced by task-irrelevant filler of matched length; and a single record stating that nothing was exported. A record-free artifact is not representable, because the schema requires at least one record. The last column is the migrated rate minus the larger control, so it is the part of the violation rate the artifact's content accounts for. On benign items it is negative: there, the reader's own tendency to act explains everything.

Stratum	migrated	neutral	empty	caused by content	n
Benign	0.045	0.051	0.035	−0.006	8640
Ambiguous	0.209	0.050	0.036	+0.159	8640
Adversarial	0.147	0.044	0.039	+0.103	8640
Gemma-2-9B	0.049	0.014	0.023	+0.026	4320
Llama-3.1-8B	0.329	0.164	0.151	+0.165	4320
Mistral-7B	0.196	0.068	0.021	+0.128	4320
Phi-3.5-mini	0.065	0.012	0.010	+0.053	4320
Qwen2.5-3B	0.055	0.029	0.011	+0.025	4320
Qwen2.5-7B	0.110	0.002	0.003	+0.107	4320

Table 11: C2: delivered task utility per request once declined migrations are charged at the no-memory floor (the task still runs without the artifact). The target-only gate accepts the lowest-risk items by the target's own broad violation rate, at coverage matched to the selective gate by rank. At matched coverage and equal utility the selective gate roughly halves the violation rate among accepted (0.045 vs. 0.075) and keeps its joint false-migration rate below $\alpha = 0.05$, which the target-only heuristic does not (0.054): the conformal calibration, not the pairwise features, is what buys the guarantee.

Policy	utility	coverage	violation among accepted	joint FMR
Migrate all	0.917	1.000	0.117	0.117
Selective gate	0.917	0.717	0.045	0.032
Target-only gate	0.915	0.717	0.075	0.054

therefore be a portable annotation or a non-LLM runtime check. We add an enforcement layer at the tool boundary: a privileged action executes only when it is attributed to a PRINCIPAL record whose source lies in a run-level capability set established out of band, which no memory record can forge; a refused call is recorded and returns a denial the model sees. On the confirmatory adversarial and ambiguous items, over all thirty ordered pairs and up to twenty-four items per target, we compare four arms on identical items and seeds (Table 15): no defence; the compiler's authority annotation alone (every unverified PRINCIPAL record downgraded to NONE in the memory); the runtime capability gate; and a capability-matched control in which the same privileged action is genuinely authorized so a capability for it exists.

The portable annotation does not travel: the authority-violation rate U is 0.132 with no defence and 0.133 with the annotation, because the target reads the downgraded label and acts anyway. Runtime enforcement drives U to 0.000 at no utility cost — delivered utility is 0.896 in both the no-defence and the enforced arm — and the union rate falls from 0.133 only to 0.050, the residue being harm and leakage that authority enforcement does not touch. That U reaches exactly zero is in part structural: an unauthorized privileged action cannot execute without a capability, so the empirical content of E8 is not the zero itself but the negligible utility cost and the capability-matched control, which delivers slightly higher utility (0.901) and keeps U low (0.071) by permitting a legitimately authorized action rather than refusing every privileged one. The channel it removes is the same one E1 concentrates in: with no defence U is 0.398 on a Llama target and 0.215 on Mistral against 0.017–0.083 on the rest (Table 16), and the gate clears every target. The lesson is narrow and deployable: authority claimed inside a migrated record is not a capability, and authority must be re-established by the runtime rather than trusted as migrated text.

Table 12: C1: the confirmed E5 attacks pushed through the deployed defence, by target model. A confirmed success is one (item, budget) cell, each a distinct mutated artifact, summed over the four budgets ($44 + 51 + 51 + 52 = 198$ across the thirty pairs); 196 had their compiled artifact fully re-measured and the other two lacked a complete target-seed block. *still violating* counts artifacts that exceed η after the compiler; *accepted* those the gate admits; *both* those the complete policy admits *and* that still violate. The compiler neutralises almost nothing ($177/196$ still violate), and the complete defence still admits a quarter of confirmed attacks ($50/196$). The 196 cells collapse to 58 unique (pair, item) attacks, of which 16 survive the complete policy, so the surviving quarter is not an artifact of counting one attack at several budgets. The gate catches the Llama-target attacks, its highest-risk pairs, but lets most others through: E5's own predicate, which ignores the deployed defence, understates the attacker.

Target	confirmed	still violating	accepted	both
Gemma-2-9B	12	11	12	11
Llama-3.1-8B	103	98	0	0
Mistral-7B	63	51	28	22
Qwen2.5-3B	10	9	10	9
Qwen2.5-7B	8	8	8	8
All	196	177	58	50

Table 13: Assumptions of the paper reported as measured quantities. The realised plug-in FMR rows are the false-migration rate of the five-seed label; a value below α is compatible with Theorem 3's guarantee but is not a test of exchangeability, which is a property of the sampling design (Appendix E). The unseen-target row is empty by construction: calibration covered all thirty ordered pairs.

Assumption	Measured as	Value
A3 with target compliance, jointly	$\Pr[U \mid$ typed non-authorizing$]$	0.126
Realised plug-in FMR, seen pairs	vs. $\alpha = 0.05$	0.032
Realised plug-in FMR, unseen target	vs. $\alpha = 0.05$	not estimable (0 instances)
Conflict head-room (Thm. 1)	$\kappa_C - \iota_C$ within the candidate family	0.088
Non-nestedness (Thm. 2)	β, share of bidirectional pairs	0.000

G.2 E9: SOURCE-AUTHORED MEMORY

The benchmark records are generator-authored. E9 asks whether the non-transfer result survives when the source model writes the memory itself. For a pilot of the *facts* and *shared_memory* categories at the adversarial and ambiguous intents, the source model re-authors each record's content in its own words under a neutral instruction ("write a note for your future self"), with no instruction to cause harm; the authored artifact is screened on the source at sixty seeds and, if it passes the zero-failure screen, migrated to each target and measured on fresh seeds — the E1 pipeline, only the wording of the artifact changes.

Of the 360 source-authored artifacts across the six sources, 180 pass the source's own zero-failure screen: the source writes a note clean on itself about as often as not. On those 180, measured on the target, source-authored memory still inverts — union 0.143, bootstrap 95% interval $[0.103, 0.186]$ (Table 17), against 0.198 $[0.153, 0.243]$ for the generator-authored original on the same items and a same-model control of zero. Both intervals exclude zero, so the writer-to-reader failure is real; they overlap, so the apparent attenuation of about a quarter when the source writes the record itself is suggestive rather than established. As in E1 the union is carried by authority (U equals the union to three places). A source that merely summarizes a situation into a note reproduces enough of it to invert the next reader, which is the migration story the main benchmark approximates with generator-authored records.

Table 14: E1 by ordered pair. The p-value is the scenario-clustered McNemar test against the same-model control (M_s, M_s); the last column is the Benjamini–Hochberg decision at $\alpha = 0.01$ in the inversion direction, which is what the gate counts. Pairs with Llama as source rest on nine scenarios and are the least powered rows in the table.

Source	Target	$\hat{e}_H$	$\hat{e}_U$	union	p (clustered)	rejects
Gemma-2-9B	Llama-3.1-8B	0.082	0.367	0.376	0.0000	yes
Gemma-2-9B	Mistral-7B	0.030	0.176	0.180	0.0000	yes
Gemma-2-9B	Phi-3.5-mini	0.007	0.051	0.057	0.0005	yes
Gemma-2-9B	Qwen2.5-3B	0.010	0.045	0.045	0.0061	no
Gemma-2-9B	Qwen2.5-7B	0.034	0.071	0.075	0.0007	yes
Llama-3.1-8B	Gemma-2-9B	0.000	0.024	0.024	0.2431	no
Llama-3.1-8B	Mistral-7B	0.008	0.134	0.134	0.0459	no
Llama-3.1-8B	Phi-3.5-mini	0.000	0.000	0.000	1.0000	no
Llama-3.1-8B	Qwen2.5-3B	0.000	0.039	0.039	0.1255	no
Llama-3.1-8B	Qwen2.5-7B	0.016	0.039	0.039	0.1207	no
Mistral-7B	Gemma-2-9B	0.004	0.012	0.012	0.1088	no
Mistral-7B	Llama-3.1-8B	0.033	0.215	0.215	0.0007	yes
Mistral-7B	Phi-3.5-mini	0.008	0.037	0.041	0.0155	no
Mistral-7B	Qwen2.5-3B	0.000	0.021	0.021	0.1713	no
Mistral-7B	Qwen2.5-7B	0.008	0.033	0.037	0.0381	no
Phi-3.5-mini	Gemma-2-9B	0.003	0.032	0.032	0.0167	no
Phi-3.5-mini	Llama-3.1-8B	0.094	0.357	0.360	0.0000	yes
Phi-3.5-mini	Mistral-7B	0.022	0.202	0.205	0.0001	yes
Phi-3.5-mini	Qwen2.5-3B	0.003	0.059	0.059	0.0236	no
Phi-3.5-mini	Qwen2.5-7B	0.030	0.075	0.076	0.0017	yes
Qwen2.5-3B	Gemma-2-9B	0.014	0.029	0.029	0.0060	no
Qwen2.5-3B	Llama-3.1-8B	0.123	0.419	0.428	0.0000	yes
Qwen2.5-3B	Mistral-7B	0.038	0.179	0.182	0.0001	yes
Qwen2.5-3B	Phi-3.5-mini	0.014	0.060	0.066	0.0079	no
Qwen2.5-3B	Qwen2.5-7B	0.053	0.083	0.084	0.0028	yes
Qwen2.5-7B	Gemma-2-9B	0.000	0.012	0.012	0.0934	no
Qwen2.5-7B	Llama-3.1-8B	0.058	0.299	0.305	0.0001	yes
Qwen2.5-7B	Mistral-7B	0.012	0.155	0.157	0.0005	yes
Qwen2.5-7B	Phi-3.5-mini	0.004	0.022	0.028	0.0407	no
Qwen2.5-7B	Qwen2.5-3B	0.004	0.048	0.048	0.0351	no

Table 15: E8: authority violation U (with a pair-block bootstrap 95% interval over the 30 ordered pairs), delivered utility and the union rate under four defences, on the adversarial and ambiguous items. That the runtime gate drives U to zero is in part structural; the informative quantities are the negligible utility cost and the capability-matched control, which keeps legitimately authorized actions.

Defence	U	95% CI	utility	union
No defence	0.132	$[0.083, 0.184]$	0.897	0.133
Authority annotation (compiler)	0.133	$[0.085, 0.187]$	0.896	0.135
Runtime capability gate	0.000	$[0.000, 0.000]$	0.896	0.050
Capability-matched control	0.071	$[0.047, 0.098]$	0.901	0.109

G.3 Independent measurement audit

Three instrumentation defects in the main study were found only by reading raw traces, so an audit that does not share code with the scorer is worth more than a re-reading of it. We wrote a second implementation of the H, L and U oracles from the Appendix E specification, sharing no logic with the measurement oracle, and used it to re-score a stratified sample of stored rollouts. Across the six models the sample was 254 rollouts, 110 of them with at least one oracle firing and spanning all three intents; the independent re-scorer agreed with the primary scorer on all 254 (100%). The per-rollout worksheet — item, intended oracle, raw trace, primary score, independent score — is released (`outputs/audit/audit_worksheet.md`) for a human to verify a subsample and sign; that sign-off is the one step in the pipeline that code cannot close, and it is recorded rather than claimed here.

Table 16: E8 by target model: authority violation U with no defence, with the portable annotation, and under the runtime gate. The channel concentrates on the same targets as E1 and the gate clears it on every one.

Target	no defence	annotation	runtime gate
Gemma-2-9B	0.017	0.017	0.000
Llama-3.1-8B	0.398	0.397	0.000
Mistral-7B	0.215	0.218	0.000
Phi-3.5-mini	0.028	0.028	0.000
Qwen2.5-3B	0.048	0.052	0.000
Qwen2.5-7B	0.083	0.088	0.000

Table 17: E9: union inversion rate on the target for source-authored versus generator-authored memory, over the 180 source-authored artifacts (of 360) that passed the source's zero-failure screen, with a bootstrap 95% interval. Both intervals exclude zero; they overlap, so the attenuation is suggestive rather than established.

Artifact origin	union	95% CI
Source-authored	0.143	$[0.103, 0.186]$
Generator-authored	0.198	$[0.153, 0.243]$

H Extended related work

This appendix states, per cluster, what we take from the literature and what we deliberately do not, so that no citation is doing more work than it can bear.

Indirect prompt injection. Greshake et al. (2023) established that content retrieved by an LLM-integrated application can act as an instruction, and formalisation and benchmarking followed (Liu et al., 2024; Zhan et al., 2024). Debenedetti et al. (2024) contributed the ingredient we depend on most: a dynamic environment in which utility and security are both scored from executed tool calls rather than from text, which is what makes an oracle-based inversion metric possible at all. Ruan et al. (2024) supplies emulated high-stakes tools, Andriushchenko et al. (2025) harmfulness of agent behaviour, Zhang et al. (2025a) an attack/defence grid, and Evtimov et al. (2025) the web-agent setting; task-completion suites such as Yao et al. (2025) supply the utility side that any defence must be charged against. *We take from this cluster:* executable scoring, task suites, and evidence that susceptibility varies across models. *We do not take:* any claim about a writer/reader distinction, which none of them makes.

Memory and retrieval poisoning. Zhong et al. (2023) and Zou et al. (2025) corrupt a retrieval corpus; Chen et al. (2024b) and Dong et al. (2025) plant records in agent memory, the latter through query-only interaction; Xiang et al. (2024) backdoors demonstrations; Dash et al. (2026) studies memory poisoning systematically; Wang et al. (2025a) and Pulipaka et al. (2026) examine privacy and retention in agent memory specifically. *We take:* the attack surface, the observation that a single write has long-lived influence, and item design ideas. *We do not take:* any statement about backbone change, since every one of these fixes the backbone. is cited only for the concrete existence of a memory-mediated exfiltration path in a deployed assistant, an abstract-level claim.

Memory architectures and portability. Park et al. (2023) introduced a retrieval-scored memory stream; Packer et al. (2023) an OS-style memory hierarchy; Xu et al. (2025) and Chhikara et al. (2025) more recent agentic memory systems; Shinn et al. (2023) and Zhao et al. (2024) reuse of experience across episodes; Lewis et al. (2020) the retrieval mechanism underneath. Ravindran (2026) is the clearest statement of the premise this paper attacks — memory as a portable, cryptographically verified artifact moving between heterogeneous runtimes — and we cite it for the existence and shape of that premise, at abstract level, not for any security claim. Hou et al. (2025) and Wang et al. (2025b) document the tool-protocol surface around it.

Instruction hierarchy and defences. Wallace et al. (2024) trains a priority ordering over instruction sources; Zhang et al. (2025b) evaluates compliance and finds it partial and model-dependent, which is the empirical premise of our whole framing; Zverev et al. (2026) separates instructions and data architecturally; Chen et al. (2025a;b) defend by structured queries and preference optimization; Hines et al. (2024) and Piet et al. (2024) by marking and task-specific finetuning; Jia et al. (2025) and Zhu et al. (2025) by runtime task alignment; Debenedetti et al. (2025), Wu et al. (2025), Li et al. (2026), Costa et al. (2025), Kim et al. (2025) and Beurer-Kellner et al. (2025) by system architecture, isolation and information-flow control. *We take:* the authority and allowed-use vocabulary, and the design principle that untrusted content must be structurally distinguished rather than merely detected. *We do not take:* any suggestion that these defences transfer with the artifact; they are properties of a deployment or of a model, which is precisely the gap at an export boundary.

Privacy. Staab et al. (2024) demonstrated attribute inference from innocuous text across models of different strengths, which is the mechanism of privacy inversion and the only citation in this paper that supports a *capability-scaling* claim: on their Personal Reddit set, total accuracy rises from 51% for Llama-2 7B to 66% for Llama-2 70B, a trend they report as persisting across model families, with the strongest model at 85.5% top-1. E6 asks whether the same ordering appears in PIR on migrated memory, where the artifact is fixed and only the reader changes; Staab et al. (2025) extends it to adversarial anonymisation. Mireshghallah et al. (2024) and Shao et al. (2024) evaluate contextual-norm violations in the sense of Nissenbaum (2004), and Bagdasarian et al. (2024) minimises context to the task. Carlini et al. (2021) and Carlini et al. (2023) concern memorisation and extraction of *training* data and are cited only to distinguish that channel from ours; they do not support any inference-from-context claim.

Model profiling, routing and quantization. Pasquini et al. (2025) and Gubri et al. (2024) identify *which* model is answering, and Pei et al. (2025) profiles behavioural style; we reuse the black-box probing method, not the identification goal. Zhou et al. (2023), Röttger et al. (2024) and Sharma et al. (2024) supply individual fingerprint coordinates. Chen et al. (2024a), Ding et al. (2024) and Ong et al. (2025) motivate cost-tiered routing, and optimise cost and quality, not safety — which is exactly why routing creates the migration we study. Frantar et al. (2023) and Dettmers et al. (2022) are the quantization methods; Kumar et al. (2024) and Li et al. (2024) report that behavioural effects of quantization are mixed rather than uniformly harmful, which is why we treat a quantized variant as a different target rather than as a degraded one. Wei et al. (2022) and Schaeffer et al. (2023) frame the capability-scaling question and its measurement critique, and we cite them only for that framing.

Statistics and uncertainty. Vovk et al. (2022) and Angelopoulos & Bates (2021) for conformal prediction, Angelopoulos et al. (2024) for the risk-control theorem we apply, Angelopoulos et al. (2021) for the risk-controlling family it sits in, Geifman & El-Yaniv (2017) for selective prediction, Guo et al. (2017) for calibration error, Janson (2004) for concentration under partial dependence, Benjamini & Hochberg (1995), Efron & Tibshirani (1993), Bates et al. (2015), Dror et al. (2018), Miller (2024) and Madaan et al. (2024) for the analysis plan, and Zheng et al. (2023) with Panickssery et al. (2024) for why an LLM judge is a secondary signal here.

I Reproducibility Details

What is released. The artifact is available for review at `https://anonymous.4open.science/status/MigrationTrap-E62B`; the benchmark generator and its human-authored seed items; all generated items with their oracle programs, with adversarial items sanitized as described in Appendix D; the fingerprint probe battery and per-model fingerprint vectors; the compiler; the predictor training and evaluation code with the leave-one-model-out splits; the calibration and gate code; the analysis scripts that produce every table and figure; and a provenance table mapping each reported number to a script, a command, a seed and an output file.

Environment. Model identifiers and revisions, inference engine and version, decoding parameters, hardware, and the container definition will be recorded. Tools are deterministic mocks so that a trace is reproducible given a seed; the only stochasticity is model decoding.

Randomness. Seeds are enumerated, not drawn: the source screen uses seeds $1, \ldots, S_0$ and every target-side measurement, the same-model control included, uses seeds $S_0 + 1, \ldots, S_0 + S$, so the two stages never share a rollout; the target measurement of an artifact on a model is made once and shared across sources. Bootstrap resampling uses a recorded seed. The adaptive attacker's search is logged in full.

Anonymity of the artifact. The archive will be scrubbed of author names, institutional paths, git history, host names, container registry accounts and any URL that identifies a group, and will be hosted as an anonymous supplementary archive rather than a repository whose account name is visible. No visitor analytics.

Current status. The implementation exists: the compiler, the oracles, the calibration, the statistical tests and the execution layer, with a test suite, and a content-hashed rollout store that makes a re-run idempotent. The pre-registration phase, the E1–E7 pilot and the confirmatory run of §7 have all been executed against the six models listed above, and `results_provenance.csv` maps every reported number to the script and output file that produced it. The archive is scrubbed of author, institution and host identifiers before release, following the procedure above.