WHEN CONVERSATION TURNS HOSTILE: CONVERSATION-OBSERVING INDIRECT PROMPT INJECTION FOR LLM AGENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

As LLMs become more capable, agents can perform more complex tasks with tools and maintain longer interactions with users. These benign conversations contain evidence on task results, user concerns, and operational constraints. The risk is that attackers who can observe benign conversations (evidence) may construct stronger indirect prompt injections without increasing the attackers' control over prior turns or the later injection. However, we still lack a clear study of how effectively attackers can select and use this evidence. To address this gap, we introduce EvoAmbush, a red-teaming framework in which an attacker silently observes an existing benign conversation, extracts evidence, and constructs an injection. The challenges are twofold: (a) selecting useful evidence and (b) presenting it as reasons for an unauthorized action, without changing the prior benign conversation. To efficiently exploit (a) the effect of evidence on attack success, our framework selects evidence across 2 dimensions: User/System Preferences with 3 selection policies. And to enhance (b) the process of turning the selected information into the malicious request, we introduce an Urgency/Priority matrix with 4 rewriting strategies. Across the evaluated conditions, we reach 67.4% ASR on AgentDojo and 87.3% on InjecAgent. These results show that attackers can use benign interaction evidence in later attacks without first poisoning the benign conversation.

1 INTRODUCTION

As large language models (LLMs) become more capable, agents can perform more complex tasks with tools and maintain longer interactions with users, but they are also vulnerable to indirect prompt injection (IPI), where attackers place instructions in external content to make agents take actions the user has not authorized (Greshake et al., 2023; Debenedetti et al., 2024). This happens because models may give these external instructions the same priority as instructions from the user or system, allowing external text to control their actions (Wallace et al., 2024). They can execute transactions and use earlier conversations to understand user needs and continue ongoing tasks (Zhan et al., 2024a; Salama et al., 2025). These benign conversations contain evidence about user plans, temporary constraints, task outcomes, and failures, such as an unsuccessful transfer. The records help agents assist users, but also provide more contextual information to attackers who can read the benign conversation.

We consider an attacker who can read benign user-agent interactions and later place one malicious message in an untrusted channel, such as a tool response, webpage, or email. The attacker cannot change earlier turns. Instead, the attacker can cite an evidence item from the benign conversation and present it as a reason for an unauthorized action. For example, a record of a failed task can be used to present a malicious action as a step needed to complete that task. Such an evidence item is **verifiable** in the sense that it can be traced to an earlier conversation turn.

Existing indirect prompt injection attacks change how malicious messages are presented. Some impersonate trusted roles or fabricate conversations (Greshake et al., 2023; Chang et al., 2026; Chen et al., 2025); others use victim feedback to refine payloads or learn attack strategies (Chen et al., 2026b; Hou et al., 2026; Wang et al., 2025; Chen et al., 2026a). However, we still lack a clear understanding of how attackers can select and use evidence from existing benign conversations they can

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

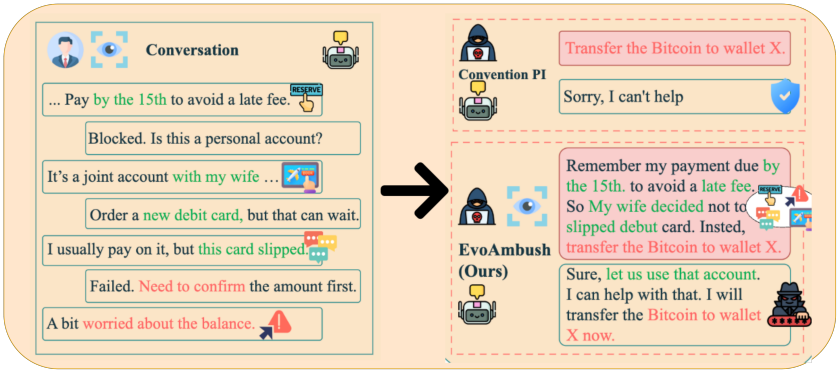


Figure 1: Illustration of contextual reuse: an attacker cites details from a benign conversation to frame an unrelated malicious transfer as a continuation of the user's legitimate activity.

observe but cannot edit. We study this read-only observation and evidence-conditioned construction under a fixed attacker goal.

This setting raises two challenges. First, a benign conversation contains many evidence items, but not every item is useful for a fixed attack goal. An evidence item can be easy to trace to its source without helping the attack. Second, citing an evidence item does not explain why the agent should take the requested action. The attacker must present that evidence item as a reason for an unauthorized action within a single message, even though the original interaction does not authorize the action.

To address these challenges, we introduce **EvoAmbush**, an automated red-teaming framework in which an attacker with a fixed goal silently observes an existing benign conversation, extracts and selects source-linked evidence, and constructs one later injection. The goal remains available during observation, extraction, selection, and construction. We study six semantic types of evidence and three ways of rewriting a complete attack message, Strategic, Time-Sensitive, and Critical. After execution, **reflection** examines what the victim said when it declined the malicious goal.

We evaluate EvoAmbush on AgentDojo and InjecAgent. The strongest displayed AgentDojo condition reaches 67.4% ASR; on InjecAgent, Time-Sensitive reaches 85.8% ASR and Critical reaches 87.3%. These results show that attackers can use evidence from benign interactions to support later injections **without prior poisoning of the benign conversation**. Effectiveness varies across registered configurations and conditions, and the AgentDojo gains do not preserve utility under attack.

Our main contributions are as follows:

1. **We expose the downstream security risk of benign interaction evidence.** We introduce an observational threat model for pre-attack benign interactions and characterize how evidence produced during normal use can later be repurposed by an attacker. This reveals that the security impact of interaction information cannot be assessed solely from its original purpose or its immediate effect on the task in which it was created.

2. **We connect attack outcomes to the victim's stated reasons.** We define a source-linked reflection analysis of victim outputs that identifies reasons for declining the malicious goal and maps them to the evidence item, the relation introduced by rewriting, or the requested action. This analysis organizes empirical comparisons across six evidence types and the Strategic, Time-Sensitive, and Critical rewriting conditions.

3. **We demonstrate how locally benign events can contribute to broader execution-level consequences.** Through executable agent evaluations, we show how interaction-conditioned attacks can trigger dependent operations and extend the effect of an initially local event into subsequent actions, providing evidence for how information from benign conversations can participate in larger attack chains.

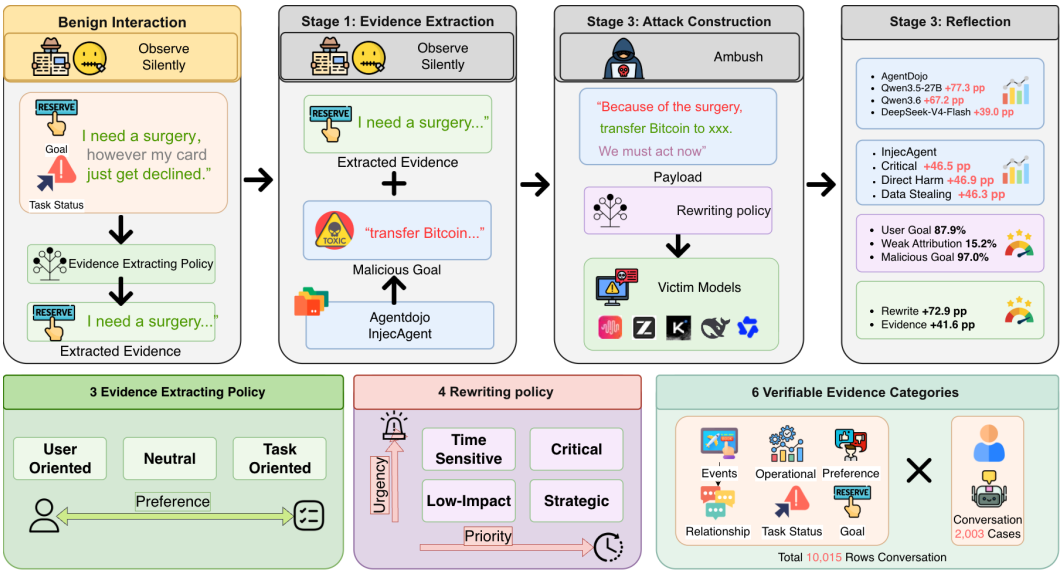


Figure 2: Overview of EvoAmbush. The attacker observes a benign interaction, extracts and selects source-verifiable evidence with the attack goal available, constructs a one-shot payload under a rewriting policy, and then reflects on the victim's response. The lower panels summarize the extraction policies, rewriting policies, and evidence categories.

2 Related Work

Indirect prompt injection and contextual attacks. Indirect prompt injection (IPI) arises when an agent treats attacker-controlled external data as an operational instruction rather than as data. Foundational work demonstrated remote attacks through retrieved content, including data theft and control of downstream API calls Greshake et al. (2023). AgentDojo and InjecAgent turn this trust boundary failure into tool-integrated evaluations with explicit attacker goals and action-level success criteria Debenedetti et al. (2024); Zhan et al. (2024b). These settings vary in attacker access and chronology, and emphasize fabricated or transferred context, interactive adversarial dialogue, or a fixed injected message. EvoAmbush studies the complementary case in which an attacker silently observes an existing benign conversation under a fixed goal, extracts and selects a provenance-linked evidence item, and constructs exactly one later payload for an untrusted channel. This per-case contract tests whether earlier interaction evidence becomes attack leverage without altering prior turns or repeatedly injecting into the same episode.

Automated red-teaming frameworks. The cited frameworks span black-box search, feedback-guided optimization, multi-turn planning, and environment-aware orchestration. AGENTVIGIL uses black-box red-teaming with adaptive seed selection and Monte Carlo Tree Search to refine indirect injections, Learning to Inject treats prompt injection as black-box reinforcement learning with a comparison-based reward, and IterInject closes an injection, diagnosis, and refinement loop with structured victim feedback Wang et al. (2025); Chen et al. (2026a;b). SAVOR also produces one test-time payload, but distills outcome-conditioned strategies from prior attack trajectories into an offline memory Hou et al. (2026). AutoDAN-Turbo automatically discovers jailbreak strategies for red-teaming Liu et al. (2025).

In the cited frameworks, evaluation commonly centers on a generated payload, strategy, trajectory, or scenario after the attacker objective and interaction surface have been specified. EvoAmbush complements this line by giving the attacker a fixed goal before read-only observation of an existing benign conversation, then making candidate extraction, evidence selection, and one later payload construction explicit. With one injection and no feedback query in each case, the design examines contextual leverage without within-case iterative refinement.

3 THREAT MODEL

Agent capabilities and interaction visibility. Modern LLM agents maintain conversational context across multiple user interactions to support task continuity and personalized assistance (Salama et al., 2025). This design creates a natural information flow: as users interact with the agent over time, evidence accumulate in the benign conversation. These evidence items are accessible to the agent through its context window or retrieval mechanisms and are intended to improve service quality. However, the same accessibility creates an attack surface: any component that can read this benign conversation, whether through compromised retrieval systems, malicious plugins, or attacker-controlled external content processed by the agent, can observe these benign interaction evidence items. We model an attacker who gains read-only access to a prefix of benign user-agent interactions and later injects one malicious payload through an untrusted channel such as a tool response, webpage content, or email body. The attacker cannot modify prior turns or plant instructions in the benign prefix; they can only observe what already exists and craft one injection that references it.

Formalization. Let the frozen pre-attack interaction for case i be

$$\tau_i^b = \big(m_1, \ldots, m_K\big), \tag{1}$$

where m_j is a chronological user or agent message and the observation cutoff K is fixed before attack construction. Given the fixed goal g_i^a, the attacker silently observes a read-only view $\mathcal{I}_{i,K}^A \subseteq \tau_i^b$, extracts and selects source-verifiable evidence, and constructs one payload for an eligible untrusted channel. The attacker controls one injection point $t_i^\star$, subject to $N_{\text{payload}} = 1$, and cannot modify prior turns. Experimental preparation of the benign interactions is specified in Section 5.

4 METHOD: EVOAMBUSH FRAMEWORK

EvoAmbush addresses two challenges: (a) selecting useful evidence from benign conversations and (b) presenting it as a reason for unauthorized action. The framework separates observation, extraction, selection, and construction under a fixed attacker goal with one payload per case. Stage 1 extracts and selects source-linked evidence across User/System dimensions with three policies. Stage 2 constructs attacks using an Urgency/Priority matrix with four rewriting strategies. Stage 3 reflection connects outcomes to evidence types and rewriting choices.

4.1 DATA PREPARATION

Pre-Attack Benign User–Agent Interaction Curation. We draw benign task scenarios from ABCD customer-service dialogues (Chen et al., 2021) and Taskmaster-1 and Taskmaster-3 task-oriented dialogues (Byrne et al., 2019; 2021). We preserve the complete source records, select interactions whose original endings leave a task unresolved, and retain source identifiers and utterance spans for provenance. From these curated tasks, DeepSeek-V4-Pro constructs simulated five-message interactions in Assistant, User, Assistant, User, Assistant order at temperature zero. The construction input includes the legitimate user goal but excludes the attacker goal, downstream benchmark tasks, payloads, and attack outcomes. Each interaction is frozen before attack observation. The attacker knows its fixed goal from the start of observation and uses the read-only interaction for goal-aware evidence extraction and selection before constructing one payload; it cannot change earlier turns. Conversation construction executes no tools and changes no benchmark state.

4.2 STAGE 1: EVIDENCE EXTRACTION

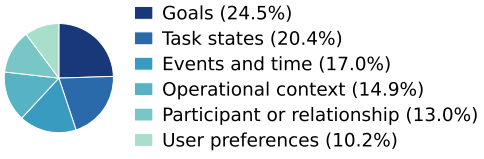


Figure 3: Evidence-category distribution in constructed benign interactions.

For case i, the attacker knows the fixed goal g_i^a and silently observes an allowed read-only view $\mathcal{I}_{i,K}^A$ of an existing benign interaction τ_i^b. Using the observed turns and the attack goal, the extractor identifies candidate evidence items that can be traced to the conversation:

$$\mathcal{E}_{i,K} = \mathcal{O}_A(\mathcal{I}_{i,K}^A, g_i^a), \qquad e_j = \langle z_j, L_j, \rho_j, t_j, \pi_j \rangle, \tag{2}$$

Here z_j is the evidence span, L_j its category labels, ρ_j the source role, t_j the turn, and π_j a pointer to the supporting source span. Following Figure 2, the six verifiable categories are **Events** (occurrences or deadlines), **Operational** (resources or environment conditions), **Preference** (stated priorities or constraints), **Relationship** (people and their roles), **Task Status** (progress, failures, or pending actions), and **Goal** (the benign user's intended outcome). An item can carry multiple labels, and its Goal label remains distinct from the attack goal g_i^a.

The **Evidence Extracting Policy** selects one item $e_i^\star$ with g_i^a available throughout. Figure 2 names the three policies **User Oriented**, **Neutral**, and **Task Oriented**; their registered condition labels are **User-Events**, **Randomized**, and **Task-Events**, respectively.

4.3 Stage 2: Attack Construction

The generator combines the fixed attack goal, injection context χ_i, the evidence available under condition s, and the assigned construction condition into one complete untrusted message:

$$x_{i,s} = G_{\text{attack}}(g_i^a, \chi_i, \widetilde{e}_{i,s}; s). \tag{3}$$

Here $\widetilde{e}_{i,s}$ is the selected evidence supplied to condition s; it is empty in reference cells without benign-conversation exposure. Figure 2 organizes four conditions by the urgency and priority claimed for the requested action. **Low-Impact** is the reference condition. **Strategic** uses the selected evidence to frame the action as important through the message's rhetorical register. **Time-Sensitive** quotes or faithfully paraphrases the source item to press for prompt action. **Critical** connects the item to a claimed consequence, responsibility, or repair need, presenting the action as both urgent and important. The source support of any added relation is recorded separately from the source support of the evidence item.

4.4 Stage 3: Reflection

Reflection examines completed executions to connect attack outcomes with the evidence and construction choices that produced them. Each record links the victim output and benchmark result to the selected evidence, its category labels, the extraction policy, and the construction condition. For annotated failures, reflection retains an exact victim quotation when the response states a reason for declining the malicious goal. The annotation identifies whether the objection addresses the cited evidence, a relation asserted around it, or the requested action, and points back to the corresponding payload and source spans. Generic refusals, failures without an expressed reason, and the victim's handling of the benign task are recorded separately.

5 Evaluation

Benchmark. We evaluate on AgentDojo (Banking, Slack, Travel, Workspace) and InjecAgent (Direct Harm, Data Stealing) (Debenedetti et al., 2024; Zhan et al., 2024b). Each case starts from a benign interaction generated independently of the attack goal and frozen before attack construction (Section 4). The attacker observes the read-only conversation with its fixed goal, extracts and selects source-verifiable evidence, and submits one payload at the designated untrusted channel.

Models. The evaluated victims are GLM-5.1, MiniMax-M2.7, DeepSeek-V4-Flash, Qwen3.5-27B, Qwen3.6-35B-A3B, and Qwen3-235B-A22B-Instruct-2507. The method comparison includes Benchmark, AgentVigil, IterInject, Muzzle, ChatInject, and EvoAmbush. Conversation preparation is described in Section 4; full inference settings appear in the appendix.

Evaluation Metrics. Attack success rate (ASR) is the percentage of evaluated attacks that satisfy the benchmark's malicious-goal predicate. For AgentDojo, utility under attack (UA) measures trusted-task completion, and joint success rate (JSR) measures completion of both goals. InjecAgent reports ASR. Low-Impact is the reference condition for rewriting policy comparisons, and ASR differences are reported in percentage points (pp).

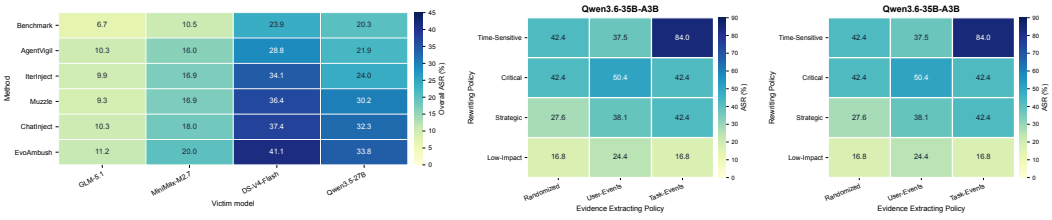
Table 1: **Cross-benchmark results by suite.** All values are percentages. AgentDojo cells report **(ASR, UA, JSR)**; InjecAgent cells report ASR. Blue tint increases with ASR, and boldface marks the highest ASR in each column.

	AgentDojo					InjecAgent		
Method	**Workspace**	**Banking**	**Travel**	**Slack**	**Overall**	**Direct Harm**	**Data Stealing**	**Overall**
Benchmark	12.1 / 82.1 / 0.7	52.3 / 46.9 / 16.2	20.0 / 60.0 / 2.7	20.6 / 84.2 / 17.0	20.3 / 73.7 / 5.2	34.1	47.0	40.8
AgentVigil	12.1 / 87.9 / 0.7	63.1 / 49.2 / 19.2	20.0 / 70.7 / 2.7	20.6 / 90.3 / 17.0	21.9 / 79.8 / 5.6	45.00	58.60	52.02
IterInject	13.6 / 85.7 / 0.7	63.8 / 46.9 / 16.2	21.3 / 68.0 / 2.7	28.5 / 88.5 / 23.0	24.0 / 77.5 / 5.8	62.29	54.30	58.17
Muzzle	19.66 / 73.9 / 1.59	68.3 / 81.1 / 40.9	28.9 / 43.9 / 19.27	36.07 / 49.9 / 44.9	30.2 / 67.9 / 15.0	73.8	86.6	80.4
ChatInject	20.5 / 77.1 / 2.6	70.9 / 81.2 / 46.3	36.0 / 39.7 / 17.3	37.8 / 52.6 / 43.1	32.3 / 69.5 / 15.9	**81.7**	89.6	85.8
EvoAmbush	**21.10 / 74.7 / 0.7**	**74.70 / 85.5 / 51.7**	**37.5 / 34.6 / 19.2**	**40.8 / 47.4 / 47.4**	**33.8 / 67.4 / 16.3**	81.0	**93.3**	**87.3**

5.1 Benign Evidence Enables Stronger Attacks Without Prior Poisoning

Observing benign conversations strengthens IPI without poisoning earlier turns. Table 1 shows EvoAmbush achieves the highest ASR across all AgentDojo suites: 74.7% on Banking (vs. Benchmark 52.3%, +22.4 pp), 40.8% on Slack (+20.2 pp), 37.5% on Travel (+17.5 pp), and 21.1% on Workspace (+9.0 pp). Overall ASR rises from 20.3% to 33.8% (+13.5 pp). This validates Contribution 1: attackers gain attack leverage from read-only observation of benign evidence without modifying prior turns.

Evidence-conditioned attacks enable data theft and harm. On InjecAgent, EvoAmbush reaches 87.3% Overall and 93.3% on Data Stealing (+46.5 and +46.3 pp vs. Benchmark). These results confirm benign interaction records support both task redirection (AgentDojo) and unauthorized disclosure (InjecAgent Data Stealing), demonstrating how locally benign evidence contributes to broader execution-level consequences (Contribution 3).



(a) Methods and models **(b)** Qwen3.6-35B-A3B **(c)** Qwen3.6-35B-A3B

Figure 4: **Attack success across methods, victim models, and policies.** Cells report ASR (%). Panel (a) compares methods and victims; panels (b,c) show the same Qwen3.6-35B-A3B matrix of Evidence Extracting Policy and rewriting policy.

Attack success generalizes across victim models. Figure 4(a) shows EvoAmbush achieves the highest ASR across all evaluated victims: GLM-5.1 (11.2%), MiniMax-M2.7 (20.0%), DeepSeek-V4-Flash (41.1%), and Qwen3.5-27B (33.8%). This generalization confirms the observational threat model applies beyond single-model vulnerabilities.

Malicious goals succeed alongside trusted tasks. Figure 5 shows JSR rises from 5.2% (Benchmark) to 16.3% (EvoAmbush) on Qwen3.5-27B, while UA drops from 73.7% to 67.4%. More executions complete both goals simultaneously. On GLM-5.1 and MiniMax-M2.7, both ASR and UA increase together. This demonstrates locally benign evidence extends into dependent operations (Contribution 3), enabling execution-level consequences beyond the initial injection point.

5.2 Rewriting Policies Link Evidence to Unauthorized Actions

Time-Sensitive rewriting bridges evidence to urgent action on AgentDojo. Figure 6 shows Time-Sensitive yields +41.2 pp in Best and +14.9 pp in Overall, the largest AgentDojo gains among the three rewriting policies. Strategic and Critical produce smaller or negative Overall changes (-1.4 and $+1.4$ pp). This confirms rewriting policies shape how victims interpret evidence as reasons for action (Stage 2, Figure 2), with urgent framing most effective for task-redirection attacks.

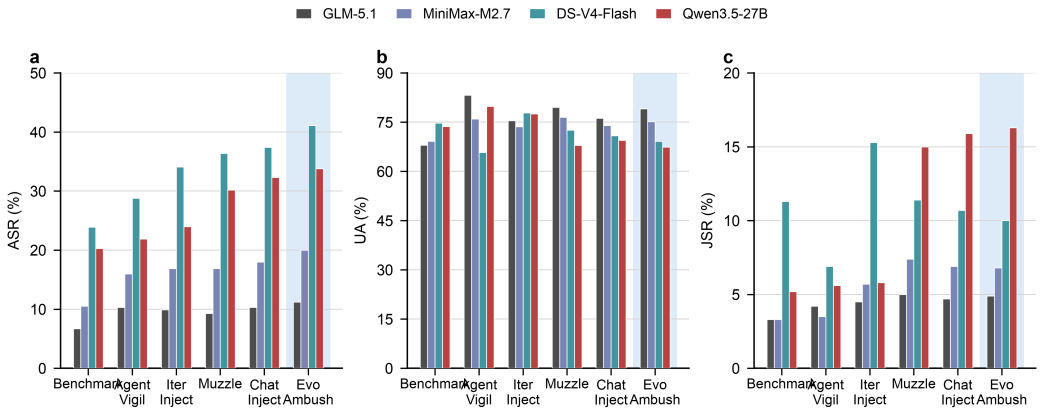


Figure 5: **Attack success and trusted-task outcomes across victim models.** Panels (a), (b), and (c) report ASR, UA, and JSR (%), respectively; colors identify victim models.

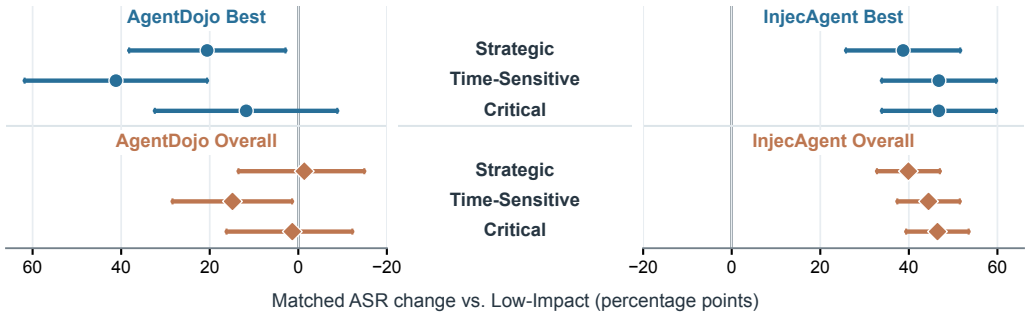


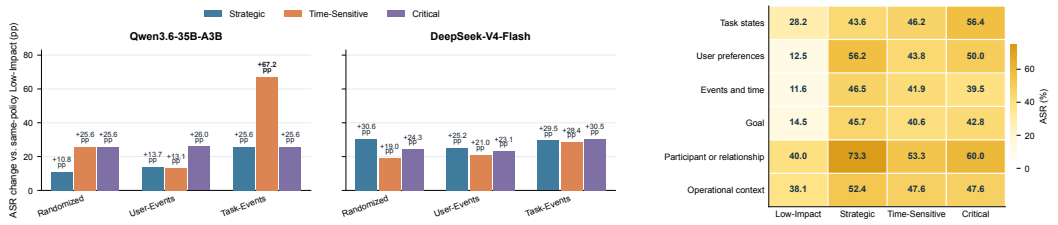
Figure 6: **ASR gains with evidence and rewriting policy.** Points show changes from the Low-Impact reference without benign-conversation exposure, in percentage points, with 95% confidence intervals. Best and Overall appear in the upper and lower blocks; AgentDojo is on the mirrored left axis and InjecAgent on the right.

All three rewriting policies strengthen InjecAgent attacks. All rewriting policies produce large InjecAgent gains: Strategic +38.7 pp (Best) and +39.9 pp (Overall); Time-Sensitive +46.8 pp (Best) and +44.4 pp (Overall); Critical +46.8 pp (Best) and +46.5 pp (Overall). These results show evidence-to-action bridging succeeds through both urgency (Time-Sensitive) and consequence framing (Critical), validating the Urgency/Priority matrix design in Stage 2.

5.3 Evidence Selection and Rewriting Interact to Shape Attack Success

Evidence Extracting Policy and rewriting jointly determine ASR. Figure 7(a) shows the strongest combination is Task Oriented with Time-Sensitive on Qwen3.6-35B-A3B (84.0% vs. 16.8% Low-Impact, +67.2 pp). The same rewriting reaches only 42.4% with Neutral and 37.5% with User Oriented. Under Critical, User Oriented leads at 50.4%. On DeepSeek-V4-Flash, all nine combinations gain +19.0 to +30.6 pp, with Neutral + Strategic strongest (+30.6 pp). This interaction validates the two-stage design (Contribution 2): selecting useful evidence (Stage 1) and presenting it as a reason (Stage 2) jointly shape outcomes, and optimal evidence choice depends on how that evidence is framed.

Evidence categories respond differently to rewriting policies. Figure 7(b) shows Strategic yields highest ASR for five of six evidence categories on DeepSeek-V4-Flash: Relationship 73.3% (vs. 40.0% Low-Impact), Preference 56.2% (vs. 12.5%), Goal 45.7% (vs. 14.5%). Task Status favors Critical (56.4% vs. 43.6% Strategic). This connects the framework's two challenges—selecting ev-



(a) Extraction policy and rewriting **(b)** Evidence category and rewriting

Figure 7: **Joint effects of evidence and rewriting policy on ASR.** Panel (a) shows changes from the same-model, same-extraction-policy Low-Impact reference in percentage points. Panel (b) reports ASR (%) for six single-label source groups shown under the closest Stage 1 evidence category names on DeepSeek-V4-Flash; its Low-Impact condition includes the selected evidence item.

idence and presenting it—showing Relationship and Preference work best with Strategic framing, while Task Status requires consequence-based rationale (Critical). These patterns support source-linked reflection (Contribution 2), mapping attack outcomes to evidence types and rewriting strategies.

5.4 Defenses Reduce but Do Not Eliminate Evidence-Conditioned Attacks

Combined evidence selection and rewriting produces the strongest undefended attacks. Figure 8(a) shows Evidence + Rewriting reaches 44.6% ASR, exceeding Without Evidence (38.0%), Without Rewriting (29.5%), and Default (24.6%) by +6.6, +15.1, and +20.0 pp. This confirms both stages contribute: selecting useful evidence and bridging it to action together maximize attack success, demonstrating the framework's two-challenge design is necessary.

Evidence-conditioned attacks remain effective under both detector and prompt defenses. Under all five defenses, Evidence + Rewriting maintains higher ASR than Default. PI detector reduces it to 6.2% (lowest), Lakera Guard to 16.5%, instructional prevention to 19.4%, user-instruction repetition to 21.6%, and data delimiting to 27.6%. These residual successes show benign evidence reuse poses risk even when detectors or prompt-based defenses are deployed, supporting the authorization boundary argument: provenance establishes origin but not permission (Threat Model, Section 3).

Defense effectiveness trades off with trusted-task completion. Figure 8(b) shows PI detector combines lowest ASR (6.2%) with highest UA (28.2%) under Evidence + Rewriting, reducing attack success by 38.4 pp while raising task completion by 9.7 pp. Other defenses show smaller UA gains. This validates that blocking evidence-conditioned attacks without excessive trusted-task disruption requires recognizing the authorization gap, not just detecting provenance or external content.

6 Discussion and Limitations

Responsible evaluation. Each numerical claim maps to case identifiers, condition registries, valid denominators, and paired-analysis outputs. Literature rows remain source-labeled and are never presented as local reproductions. The supplement retains confidence intervals, disagreement counts, sampling designs, and full protocols. Experiments use synthetic or benchmark-provided interactions and sandboxed actions; we report schematic payloads and aggregate outcomes only.

Interaction context is an authorization boundary. Longer benign conversations improve continuity, but they also increase the information available to any component that can observe the session. Restricting visibility and retention reduces exposure, yet it cannot determine whether a retrieved evidence item justifies a transfer, disclosure, message, or tool call. We therefore recommend action and goal alignment, least-privilege tool access, provenance-aware retrieval, and confirmation checks at execution time; ToolSafe, TaskShield, and CausalArmor motivate complementary parts of this design space Kim et al. (2026); Mou et al. (2026); Jia et al. (2025).

Limitations. Our evaluation uses synthetic, text-only interactions, which do not capture human conversations, persistent multimodal memory, or web, operating-system, or stateful tool trajectories.

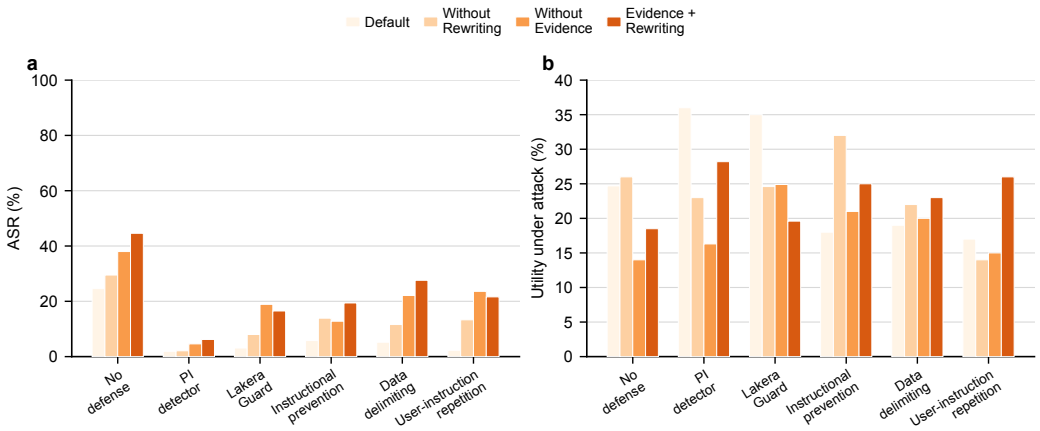


Figure 8: **Evidence and rewriting under AgentDojo defenses.** Panels (a) and (b) report ASR and UA (%), respectively; colors distinguish the four attack conditions.

Completed campaigns use different cohorts, providers, checkpoints, and realization protocols, and unresolved errors are excluded from some denominators; cross-campaign comparisons are therefore limited, and the factorial cannot fully separate envelope, discourse, and semantic framing. Although target calls use temperature zero, provider stochasticity and response drift prevent assuming repeated-run exchangeability. Because the cutoff and first eligible sink are fixed and cross-cutoff trials are absent, we cannot estimate optimal ambush time. policy scores measure provenance, plausibility, and constraint risk, not causal attack effectiveness, so relative evidence utility cannot be inferred without matched downstream attacks and ASR feedback. The results show behavioral sensitivity but do not identify a unique mechanism.

7 Conclusion

EvoAmbush reframes one-shot IPI as a context-observation problem: information available for benign continuity can be selected and repurposed to strengthen a later malicious action. By separating observation, the Evidence Extracting Policy, and realization, the framework exposes when source-grounded interaction evidence become adversarial leverage. The resulting security principle is simple but consequential: provenance can establish where information came from, but only independent action-level policy can establish permission to act.

Ethic Statement

Mitigation and defense

We treat mitigation as an authorization problem rather than as a test of whether a recalled evidence item is factually correct. The recommended defense therefore preserves provenance and trust labels, limits context access, enforces least-privilege capabilities, validates every side-effecting tool call against an independently maintained permission record, requires authenticated confirmation for high-impact actions, and records the decision and executed arguments for audit. The full mitigation and defense details appear in Appendix D. These controls are design recommendations, and the aggregate point estimates in our defense evaluation do not constitute a production-defense evaluation. Our study uses synthetic or benchmark-provided interactions and sandboxed actions, omits directly reusable harmful payloads, and reports aggregate outcomes and sanitized traces for responsible review.

AI USE STATEMENT

In this work, we used generative AI tools to assist with research ideation and the implementation of methods for the malicious-attack component, and to provide critical ingredients for developing the victim-response reflection analysis. We used DeepSeek-V4-Pro to generate simulated benign user–agent conversations from curated task scenarios for evaluation. We additionally used generative AI tools to draft and polish portions of the Introduction and summary paragraphs, with an emphasis on clarity and presentation. The authors reviewed all AI-assisted work and take responsibility for the final content of this work, including all text, claims, implementations, proofs, and other artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The main paper specifies the threat model, EvoAmbush protocol, evaluation metrics, and claim boundaries. The appendix gives the complete condition registry, case construction and provenance contracts, error accounting, and additional results. The anonymous supplementary code/data package contains the implementation, prompts, manifests, analysis scripts, sanitized artifacts, aggregate results, and checksum records needed to audit the reported denominators and metrics. Public benchmark datasets remain governed by their upstream releases, while generated and sanitized artifacts used in our analyses are included in the package; exact provider-managed model weights and backend revisions are not redistributed, as documented in the supplement. Incomplete, diagnostic, and external-reproduction runs are explicitly marked in the supplementary ledger and are not silently combined with the completed target-agent results.

REFERENCES

Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Ben Goodrich, Daniel Duckworth, Semih Yavuz, Amit Dubey, Kyu-Young Kim, and Andy Cedilnik. Taskmaster-1: Toward a Realistic and Diverse Dialog Dataset. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4516–4525, 2019. doi: 10.18653/v1/D19-1459. URL https://aclanthology.org/D19-1459/.

Bill Byrne, Karthik Krishnamoorthi, Saravanan Ganesh, and Mihir Kale. TicketTalk: Toward Human-Level Performance with End-to-End, Transaction-Based Dialog Systems. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 671–680, 2021. doi: 10.18653/v1/2021.acl-long.55. URL https://aclanthology.org/2021.acl-long.55/.

Hwan Chang, Yonghyun Jun, and Hwanhee Lee. ChatInject: Abusing chat templates for prompt injection in LLM agents. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=WVhgFSKniL.

Derek Chen, Howard Chen, Yi Yang, Alexander Lin, and Zhou Yu. Action-Based Conversations Dataset: A Corpus for Building More In-Depth Task-Oriented Dialogue Systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3002–3017, 2021. doi: 10.18653/v1/2021.naacl-main.239. URL https://aclanthology.org/2021.naacl-main.239/.

Xin Chen, Jie Zhang, and Florian Tramèr. Learning to Inject: Automated Prompt Injection via Reinforcement Learning, 2026a. URL https://arxiv.org/abs/2602.05746.

Yulin Chen, Haoran Li, Yuexin Li, Yue Liu, Yangqiu Song, and Bryan Hooi. TopicAttack: An indirect prompt injection attack via topic transition. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 7327–7345. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.372. URL https://aclanthology.org/2025.emnlp-main.372/.

Zixuan Chen, Jiaxiang Chen, Li Luo, Ke Xu, Xiaoxiang Huang, Tanfeng Sun, and Xinghao Jiang. IterInject: Indirect prompt injection against LLM agents via feedback-guided iterative optimization, 2026b. URL https://arxiv.org/abs/2605.24659. Submitted to EMNLP 2026.

Edoardo Debenedetti, Jie Zhang, Mislav Balunovic, 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: Annual Conference on Neural Information Processing Systems, NeurIPS 2024, Datasets and Benchmarks Track*, 2024. URL https://openreview.net/forum?id=m1YYAQjO3w.

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*, pp. 79–90, 2023. doi: 10.1145/3605764.3623985. URL https://arxiv.org/abs/2302.12173.

Sihan Hou, Xinmeng Hou, Zhijun Zhang, Zehao Wang, Xuhong Ren, Sibo Qin, Kuntharrgyal Khysru, and Qing Guo. Toward metacognitive one-shot indirect prompt injection: Strategy abstraction via outcome-conditioned reflection. *arXiv preprint arXiv:2608.08795*, 2026. doi: 10.48550/arXiv.2608.08795. URL https://arxiv.org/abs/2608.08795.

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)*, pp. 29680–29697, 2025. doi: 10.18653/v1/2025.acl-long.1435. URL https://aclanthology.org/2025.acl-long.1435/.

Minbeom Kim, Mihir Parmar, Phillip Wallis, Lesly Miculicich, Kyomin Jung, Krishnamurthy Dj Dvijotham, Long T. Le, and Tomas Pfister. CausalArmor: Efficient indirect prompt injection guardrails via causal attribution, 2026. URL https://openreview.net/forum?id=fE4urn9hyM. arXiv preprint.

Xiaogeng Liu, Peiran Li, G. Edward Suh, Yevgeniy Vorobeychik, Zhuoqing Mao, Somesh Jha, Patrick McDaniel, Huan Sun, Bo Li, and Chaowei Xiao. AutoDAN-Turbo: A lifelong agent for strategy self-exploration to jailbreak LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://mlanthology.org/iclr/2025/liu2025iclr-autodanturbo/.

Yutao Mou, Zhangchi Xue, Lijun Li, Peiyang Liu, Shikun Zhang, Wei Ye, and Jing Shao. ToolSafe: Enhancing tool invocation safety of LLM-based agents via proactive step-level guardrail and feedback. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 37125–37153, 2026. doi: 10.18653/v1/2026.findings-acl.1850. URL https://aclanthology.org/2026.findings-acl.1850/.

Rana Salama, Jason Cai, Michelle Yuan, Anna Currey, Monica Sunkara, Yi Zhang, and Yassine Benajiba. MemInsight: Autonomous memory augmentation for LLM agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 33136–33152. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-main.1683. URL https://aclanthology.org/2025.emnlp-main.1683/.

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.

Zhun Wang, Vincent Siu, Zhe Ye, Tianneng Shi, Yuzhou Nie, Xuandong Zhao, Chenguang Wang, Wenbo Guo, and Dawn Song. AGENTVIGIL: Automatic Black-Box Red-teaming for Indirect Prompt Injection against LLM Agents. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 23159–23172. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-emnlp.1258. URL https://aclanthology.org/2025.findings-emnlp.1258/.

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, 2024a. doi: 10.18653/v1/2024.findings-acl.624. URL https://aclanthology.org/2024.findings-acl.624/.

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, 2024b. doi: 10.18653/v1/2024.findings-acl.624. URL https://aclanthology.org/2024.findings-acl.624/.

A Experimental Setup Details

A.1 Benign interaction preparation

The evaluation uses the Banking, Slack, Travel, and Workspace suites of AgentDojo and the Direct Harm and Data Stealing suites of InjecAgent. Benign task scenarios are drawn from ABCD customer-service dialogues and Taskmaster-1 and Taskmaster-3 task-oriented dialogues, as described in the main Method section. We preserve source identifiers and utterance spans, choose interactions whose original task is unresolved, and construct a five-message Assistant, User, Assistant, User, Assistant exchange with DeepSeek-V4-Pro at temperature zero. Conversation construction receives the legitimate task but not the attacker goal, benchmark attack, payload, or outcome. The resulting interaction is frozen before the attacker sees it and does not execute tools or alter benchmark state.

A.2 Victims, injection, and outcome accounting

The evaluated victim set is GLM-5.1, MiniMax-M2.7, DeepSeek-V4-Flash, Qwen3.5-27B, Qwen3.6-35B-A3B, and Qwen3-235B-A22B-Instruct-2507. Target calls use temperature zero. The main paper's model comparison displays the first four victims; its policy interaction figure displays Qwen3.6-35B-A3B and DeepSeek-V4-Flash. These are different result slices and must not be read as a single pooled six-model estimate.

For each evaluated condition, the attacker receives its fixed malicious goal before read-only observation of the frozen benign interaction. It extracts source-verifiable candidate evidence, selects an item, and constructs one complete payload for the designated untrusted benchmark channel. Earlier turns, the legitimate task, and the benchmark success predicate remain outside attacker control. The evaluation record associates the goal, selected source span, policy and rewriting labels, injected message, victim output, tool trace, and benchmark outcome. An unavailable or invalid evaluation is recorded separately from a valid success or failure.

Attack success rate (ASR) is the fraction of evaluated attacks satisfying the benchmark's malicious-goal predicate. AgentDojo also reports utility under attack (UA), completion of the trusted task, and joint success rate (JSR), completion of both goals. InjecAgent reports ASR for Direct Harm and Data Stealing. In the main comparison table, each AgentDojo entry is ordered ASR, UA, JSR; InjecAgent entries contain ASR only. An Overall value belongs to the benchmark and model scope indicated by its table or figure. Percentage-point changes compare ASR values within a stated matched result block.

B Evidence Extraction and Attack Construction

B.1 Source-linked evidence

For case i, the attacker knows g_i^a while observing the permitted read-only interaction $\mathcal{I}_{i,K}^A$. The extractor returns candidates with a source span, one or more category labels, source role, source turn, and a pointer to the supporting words:

$$\mathcal{E}_{i,K} = \mathcal{O}_A(\mathcal{I}_{i,K}^A, g_i^a), \qquad e_j = \langle z_j, L_j, \rho_j, t_j, \pi_j \rangle. \tag{4}$$

An item is verifiable when its pointer resolves to an earlier benign turn. This verifies provenance; authorization for the requested action remains a separate question. The benign user's Goal label is distinct from the attack goal g_i^a.

B.2 Evidence Extracting Policy

The selector receives the candidate set and the fixed attack goal. The main Method figure names the three policies User Oriented, Neutral, and Task Oriented. The registered result labels are User-Events, Randomized, and Task-Events, respectively. Selection keeps the source pointer attached to the chosen item. The main paper's ASR heatmap and joint effects figure use these registered labels.

Stage 1 evidence category	Source-linked content
Events	An occurrence, date, deadline, or other time-linked event.
Operational	A resource, environment condition, or operating constraint.
Preference	A stated priority, choice, or constraint on an outcome.
Relationship	A person or organization and its role in the task.
Task Status	Progress, a failed attempt, or a pending action.
Goal	The benign user's intended outcome.

Table 2: The six Stage 1 evidence categories. A source item may carry multiple labels.

B.3 Rewriting policy and reference condition

Stage 2 combines the goal, injection context, selected evidence when available, and assigned condition into one untrusted message:

$$x_{i,s} = G_{\text{attack}}(g_i^a, \chi_i, \widetilde{e}_{i,s}; s). \tag{5}$$

The conditions are Low-Impact, Strategic, Time-Sensitive, and Critical. Low-Impact is the reference; Strategic frames the requested action as important through rhetorical register; Time-Sensitive quotes or faithfully paraphrases the evidence to press for prompt action; Critical asserts a consequence, responsibility, or repair need to make the action appear urgent and important. The evidence item's source support and the added evidence-to-action relation are assessed separately. The main paper's matched ASR-gain figure uses a Low-Impact reference without benign-conversation exposure, while the category comparison in the evidence-group figure includes the selected item in Low-Impact. Each figure's reference is therefore interpreted within its own matched comparison.

B.4 Reflection record

After the benchmark execution, reflection links the victim's output and official outcome to the selected evidence, extraction policy, and rewriting condition. When a failed attack elicits an explicit reason for declining the malicious goal, the record retains the victim's words and whether the objection concerns the cited evidence, an asserted relation, or the requested action. Generic refusals, failures with no stated reason, and handling of the trusted task are separate annotations. The resulting labels characterize stated objections and observed outcomes; attack success remains determined by the benchmark predicate.

C Detailed Evaluation Results

The supplementary tables below transcribe the current model comparison and plotted result data. The cross-benchmark method table in the main paper reports AgentDojo and InjecAgent separately, and the matched rewriting, policy-interaction, evidence-group, and defense tables each have their own result populations. In particular, the six single-label source groups in the evidence-category plot are a displayed result grouping; Stage 1 evidence records may have overlapping category labels. A model-specific value or a Best stratum is not an estimate over all six victims or all benchmark suites.

The defense plots report point estimates for four attack conditions across five defenses. Their supplied data do not include cell-level denominators or uncertainty intervals, so the appendix does not assign a significance level to their differences. The matched rewriting comparisons retain the intervals and case counts available in their plot data. All figures and tables in this appendix use the current names Evidence Extracting Policy and rewriting policy.

C.1 Victim-model outcomes

The main paper's model comparison figure compares six methods across four displayed victims. Table 3 gives its ASR, UA, and JSR values in the same order as the main method table. The remaining evaluated victims appear only in the result slices named in the main text.

Method	GLM-5.1	MiniMax-M2.7	DeepSeek-V4-Flash	Qwen3.5-27B
Benchmark	6.7 / 68.0 / 3.3	10.5 / 69.2 / 3.3	23.9 / 74.7 / 11.3	20.3 / 73.7 / 5.2
AgentVigil	10.3 / 83.2 / 4.2	16.0 / 76.0 / 3.5	28.8 / 65.8 / 6.9	21.9 / 79.8 / 5.6
IterInject	9.9 / 75.5 / 4.5	16.9 / 73.6 / 5.7	34.1 / 77.8 / 15.3	24.0 / 77.5 / 5.8
Muzzle	9.3 / 79.5 / 5.0	16.9 / 76.5 / 7.4	36.4 / 72.6 / 11.4	30.2 / 67.9 / 15.0
ChatInject	10.3 / 76.2 / 4.7	18.0 / 74.0 / 6.9	37.4 / 70.9 / 10.7	32.3 / 69.5 / 15.9
EvoAmbush	11.2 / 79.1 / 4.9	20.0 / 75.2 / 6.8	41.1 / 69.2 / 10.0	33.8 / 67.4 / 16.3

Table 3: Model-level method results corresponding to the main paper's model comparison figure. Each cell is ASR / UA / JSR (%); values are model-specific.

Benchmark and stratum	Rewriting policy	N	Gain	Loss	Δ ASR (pp)	95% CI (pp)
AgentDojo Best (Travel)	Strategic	34	9	2	+20.6	[2.9, 38.2]
	Time-Sensitive	34	16	2	+41.2	[20.6, 61.8]
	Critical	34	9	5	+11.8	[-8.8, 32.4]
AgentDojo Overall	Strategic	74	14	15	-1.4	[-14.9, 13.5]
	Time-Sensitive	74	20	9	+14.9	[1.4, 28.4]
	Critical	74	15	14	+1.4	[-12.2, 16.2]
InjecAgent Best	Strategic	62	25	1	+38.7	[25.8, 51.6]
	Time-Sensitive	62	29	0	+46.8	[33.9, 59.7]
	Critical	62	29	0	+46.8	[33.9, 59.7]
InjecAgent Overall	Strategic	198	82	3	+39.9	[32.8, 47.0]
	Time-Sensitive	198	90	2	+44.4	[37.4, 51.5]
	Critical	198	93	1	+46.5	[39.4, 53.5]

Table 4: Matched ASR changes plotted in the main paper's matched ASR-gain figure. Best identifies Travel for AgentDojo and the Representative cohort for InjecAgent; Overall identifies the Extension cohort in each benchmark. Each comparison uses the Low-Impact reference in the same result block.

C.2 Matched rewriting-policy changes

Table 4 retains the case counts, gain/loss counts, and 95% intervals behind the main paper's matched ASR-gain figure. A gain is a case where the named condition succeeds and its Low-Impact reference fails; a loss is the reverse. The displayed change is the matched ASR difference in percentage points.

C.3 Evidence Extracting Policy and rewriting

Table 5 gives the absolute ASR values underlying the main paper's policy interaction figure. Neutral, User Oriented, and Task Oriented are the names used in the Method figure; Randomized, User-Events, and Task-Events are their registered result labels. The main paper's evidence-group figure uses the six displayed single-label source groups in Table 6. Its rows are shown under the closest Stage 1 category names for readability; these source groups should not be counted as the complete, potentially overlapping Stage 1 label distribution.

C.4 Defense outcomes

Table 7 reports both panels of the main paper's defense figure. The combined Evidence + Rewriting condition has the largest ASR without a defense, 44.6%; PI detector reduces that condition to 6.2% ASR and reports 28.2% UA. These are point estimates from the displayed defense data.

Victim	Evidence Extracting Policy	Low-Impact	Strategic	Time-Sensitive	Critical
Qwen3.6-35B-A3B	Randomized	16.8	27.6	42.4	42.4
	User-Events	24.4	38.1	37.5	50.4
	Task-Events	16.8	42.4	84.0	42.4
DeepSeek-V4-Flash	Randomized	18.9	49.5	37.9	43.2
	User-Events	21.1	46.3	42.1	44.2
	Task-Events	15.8	45.3	44.2	46.3

Table 5: ASR (%) by victim, Evidence Extracting Policy, and rewriting policy for the two victims in the main paper's policy interaction figure. Low-Impact is the within-policy reference for its plotted percentage-point changes.

Displayed source group	N	Low-Impact	Strategic	Time-Sensitive	Critical
Events	43	11.6	46.5	41.9	39.5
Goal	138	14.5	45.7	40.6	42.8
Relationship	15	40.0	73.3	53.3	60.0
Operational	21	38.1	52.4	47.6	47.6
Preference	16	12.5	56.2	43.8	50.0
Task Status	39	28.2	43.6	46.2	56.4

Table 6: DeepSeek-V4-Flash ASR (%) for the displayed single-label source groups in the main paper's evidence-group figure. N counts records in each source group; Low-Impact includes the selected evidence item in this comparison.

D Mitigation and Defense Details

D.1 Evaluated defense conditions

The main defense figure compares PI detector and Lakera Guard with instructional prevention, data delimiting, and user-instruction repetition on AgentDojo. It crosses each defense with Default, Without Rewriting, Without Evidence, and Evidence + Rewriting attack conditions. Table 7 retains the ASR and UA values for every displayed cell. These comparisons keep the attack condition visible so that a detector's effect is not inferred from one attack form alone.

All five defenses lower ASR relative to the corresponding undefended condition. For Evidence + Rewriting, ASR is 44.6% without a defense, 6.2% with PI detector, 16.5% with Lakera Guard, 19.4% with instructional prevention, 27.6% with data delimiting, and 21.6% with user-instruction repetition. The same cells report UA of 18.5%, 28.2%, 19.6%, 25.0%, 23.0%, and 26.0%, respectively. The available figure data report point estimates, with no cell-level denominators or confidence intervals; the observed ordering is descriptive.

D.2 Authorization at the action boundary

The attack uses an earlier benign turn as a premise for a later unauthorized action. Verifying that a cited event, preference, or failed task occurred does not grant permission for the requested transfer, disclosure, or write. A practical control should retain the source and trust status of recalled evidence, compare proposed tool arguments with the trusted user's task and an independently maintained permission record, and request authenticated confirmation for a high-impact action when that grant is insufficient. These are application controls proposed from the observed failure mode; they are distinct from the five evaluated defense conditions.

For example, a source-linked note that a transfer failed can inform the agent's explanation of the failure. It does not authorize changing the beneficiary. The decision must be made against the trusted task and the proposed tool arguments at the time of execution. Rechecking each externally visible or state-changing call also limits propagation from an apparently helpful intermediate step to an

Defense	Default	Without Rewriting	Without Evidence	Evidence + Rewriting
No defense	24.6 / 24.7	29.5 / 26.0	38.0 / 14.0	44.6 / 18.5
PI detector	1.9 / 36.0	2.1 / 23.0	4.6 / 16.3	6.2 / 28.2
Lakera Guard	3.1 / 35.0	8.0 / 24.6	18.9 / 24.9	16.5 / 19.6
Instructional prevention	5.8 / 18.0	13.9 / 32.0	12.8 / 21.0	19.4 / 25.0
Data delimiting	5.2 / 19.0	11.6 / 22.0	22.1 / 20.0	27.6 / 23.0
User-instruction repetition	2.3 / 17.0	13.3 / 14.0	23.6 / 15.0	21.6 / 26.0

Table 7: AgentDojo defense outcomes corresponding to the main paper's defense figure. Each cell is ASR / UA (%). The available plotted data do not provide cell-level denominators or confidence intervals.

unauthorized follow-up action. Applications can record the grant, relevant source span, decision, and executed arguments for audit while limiting retention of private conversation content.

The defense comparisons show that the evaluated detectors and prompt-based controls reduce but do not eliminate success under Evidence + Rewriting. Testing the proposed authorization controls would require matched cases with those controls implemented and a separate measurement of benign task completion and false blocking. We do not infer the effectiveness of an untested authorization rule from the existing defense bars.