

Harnesses de seguridad para modelos generativos y sistemas agénticos entre enero de 2023 y agosto de 2026: revisión sistemática de literatura y síntesis focal de 50 estudios

Resumen

Esta revisión sistemática de literatura analiza la investigación sobre Harnesses de seguridad para modelos generativos y sistemas agénticos entre enero de 2023 y agosto de 2026. El protocolo identificó 2440 registros, consolidó 68 duplicados antes del cribado, evaluó 143 textos completos en PDF e incluyó 116 estudios en el corpus final. A partir de esos 116 estudios incluidos, se definió una síntesis focal de 50 estudios para la comparación intensiva. Los 66 estudios restantes siguieron dentro del corpus de revisión como perímetro contextual elegible: delimitan el campo y conservan trazabilidad, pero no alcanzaron el umbral combinado de ajuste, calidad, representatividad y densidad extractiva que exige la comparación intensiva. Por eso se reportan como contexto auditable y no como evidencia fina del N focal. La pregunta de investigación fue: ¿Qué arquitecturas, controles y estrategias de evaluación de los harnesses de seguridad para modelos generativos y sistemas agénticos reducen de forma más consistente los ataques de prompt injection y jailbreak, el uso inseguro de herramientas, la fuga de datos y las violaciones de políticas, y bajo qué amenazas, diseños comparativos y costes superan a las alternativas o baselines? La síntesis focal contiene 50 estudios. Los 49 estudios empíricos del subconjunto focal se clasifican, por auto-declaración metodológica y lectura del texto completo, en 2 cuantitativos y 47 experimentales o evaluativos. La síntesis separa amenaza, superficie, arquitectura de control, punto de aplicación, atacante, baseline, eficacia, falsos positivos, utilidad, latencia, coste, robustez y fallo residual. La contribución del artículo es ofrecer una síntesis reproducible basada en DOI público, PDF local, extracción estructurada, matriz de selección y anexos auditables, no solo una narración agregada de la literatura. La reproducibilidad se refiere a decisiones, matrices y artefactos derivados; los PDFs fuente solo pueden redistribuirse cuando su licencia lo permite.

Abstract

This systematic literature review examines research on security harnesses for generative models and agentic systems between January 2023 and August 2026. The protocol identified 2440 records, consolidated 68 duplicates before screening, assessed 143 full texts in PDF, and included 116 studies in the final review corpus. From that corpus, an operational focal synthesis of 50 studies was defined. The remaining 66 included studies were retained as an eligible contextual perimeter: they delimit the field and preserve traceability, but did not meet the combined threshold of fit, quality, representativeness, and extraction density required for the deepest comparison. They are therefore reported as auditable context rather than as fine-grained focal evidence. The guiding research question was: Which architectures, controls, and evaluation strategies for security harnesses protecting generative models and agentic systems most consistently reduce prompt injection and jailbreak attacks, unsafe tool use, data leakage, and policy violations, and under which threats, comparative designs, and costs do they outperform alternatives or baselines? The focal synthesis contains 50 studies. The 49 empirical studies in the focal subset are classified, based on methodological self-description and full-text reading, as 2 quantitative and 47 experimental or evaluative. The contribution is a reproducible full-text synthesis grounded in public DOI traceability, local PDF evidence, structured extraction, a visible selection matrix, and auditable appendices. Reproducibility applies to decisions, matrices, and derived artifacts; source PDFs may only be redistributed when their licenses permit it.

Palabras clave

Harnesses de seguridad para modelos generativos y sistemas agénticos, revisión sistemática, texto completo, síntesis focal, evidencia, método, calidad metodológica, revisión sistemática de literatura

Keywords

Harnesses de seguridad para modelos generativos y sistemas agénticos, systematic review, full text, focal synthesis, evidence, method, methodological quality, systematic literature review

Introducción

Los modelos generativos dejan de ser componentes pasivos cuando recuperan documentos, mantienen memoria, llaman herramientas, escriben en sistemas externos o coordinan otros agentes. En ese momento, la seguridad ya no depende únicamente de la respuesta lingüística del modelo: depende de quién introduce datos, qué instrucciones conservan autoridad, qué capacidades puede ejecutar el sistema y qué daño sigue siendo posible después de una evasión (L. Zhao et al., 2026; Louck, 2026; Yung et al., 2025; Zhang et al., 2026).

La literatura ha respondido con filtros, guardrails, clasificadores, separación entre instrucciones y datos, controles de procedencia, autorización de herramientas, sandboxing, monitores de trayectoria y verificadores de salida. Sin embargo, estas soluciones no protegen la misma superficie ni interrumpen el ataque en el mismo punto. Una tasa de bloqueo aislada puede ocultar que el atacante era estático, que la alternativa comparada era débil, que la utilidad legítima cayó o que la defensa trasladó el coste a latencia, tokens o revisión humana (Ji et al., 2025; Liang et al., 2025; Sharma et al., 2025; Zhu et al., 2025).

Esta heterogeneidad obliga a definir el objeto revisado con precisión. En este artículo, un harness de seguridad es el conjunto de controles externos o acoplados al modelo que observan, restringen, verifican, contienen o recuperan la operación del sistema durante inferencia y uso. La definición incluye controles en entrada, contexto, recuperación, memoria, razonamiento, llamadas de herramientas, salida y persistencia; excluye el entrenamiento de seguridad del modelo base cuando no existe una capa operacional evaluable. **Guardrail** se usa solo para el subconjunto de controles que clasifica, filtra o aplica una política sobre entradas, trayectorias o salidas; no se trata como sinónimo de toda la arquitectura del harness.

El problema científico no consiste, por tanto, en preguntar qué producto bloquea más ataques en abstracto. Consiste en reconstruir un contrato comparativo: amenaza, capacidad del atacante, superficie protegida, mecanismo de control, punto de enforcement, baseline, reducción observada del riesgo, utilidad preservada, sobrecoste y modo de fallo residual. Dos defensas solo pueden ordenarse cuando ese contrato es suficientemente equivalente; si no lo es, sus cifras describen experimentos distintos y no una competición común.

La evaluación adversarial añade una dificultad adicional. Un conjunto fijo de prompts mide cobertura sobre ejemplos conocidos, mientras que un adversario adaptativo busca la frontera del control, cambia la formulación, desplaza la carga maliciosa a documentos o herramientas y explota las decisiones del sistema completo. Esta diferencia separa una demostración local de una afirmación de robustez operacional y explica por qué el artículo trata adaptatividad, transferencia, ablación y fallos conocidos como evidencia central, no como anexos secundarios.

La revisión adopta además una perspectiva multiobjetivo. Bloquear más no constituye una mejora si el harness inutiliza tareas legítimas, multiplica falsos positivos, introduce una latencia incompatible con el uso o desplaza el riesgo a una capa no observada. La decisión relevante es una frontera condicionada entre seguridad, utilidad, disponibilidad, coste y recuperabilidad. Esa frontera puede producir configuraciones prometedoras, pero no autoriza un ganador universal fuera de la amenaza y el entorno evaluados.

Este artículo aborda esa brecha mediante una revisión sistemática guiada por la pregunta: ¿Qué arquitecturas, controles y estrategias de evaluación de los harnesses de seguridad para modelos generativos y sistemas agénticos reducen de forma más consistente los ataques de prompt injection y jailbreak, el uso inseguro de herramientas, la fuga de datos y las violaciones de políticas, y bajo qué amenazas, diseños comparativos y costes superan a las alternativas o baselines? La unidad de análisis es el estudio publicado y leído en texto completo, no la mera presencia de palabras clave en bases bibliográficas.

La posición que se somete a prueba es que Harnesses de seguridad para modelos generativos y sistemas agénticos no debe leerse como una lista de hallazgos, sino como una matriz de comparación entre objeto, método, evidencia, límites y condiciones de transferencia.

La lógica de la revisión es deliberadamente conservadora: un estudio solo sostiene afirmaciones de síntesis cuando dispone de DOI público, PDF local legible, extracción estructurada y una justificación visible dentro de la matriz de selección. Esta regla reduce cobertura bruta, pero aumenta la auditabilidad del manuscrito final.

La contribución buscada es doble. En el plano sustantivo, el artículo identifica qué combinaciones defensivas muestran la señal comparativa mejor sustentada y bajo qué límites dejan de dominar. En el plano teórico, propone una gramática para acumular evidencia sin confundir detección, prevención, contención y recuperación. Esta gramática convierte la seguridad en una propiedad situada del sistema y obliga a que cualquier afirmación de superioridad declare también su coste y su frontera de fallo.

Marco teórico

El corpus sobre Harnesses de seguridad para modelos generativos y sistemas agénticos se interpreta como un campo de evidencia en construcción. La revisión no presupone que exista una teoría única capaz de ordenar todos los estudios; primero identifica cómo cada artículo define su objeto, qué método usa, qué unidad analiza y qué resultado afirma (Alizadeh et al., 2025; L. Zhao et al., 2026; Yung et al., 2025; Zhang et al., 2026).

Para evitar que el marco teórico sea solo un inventario de autores o conceptos, esta revisión lo organiza en tres capas. La primera capa recoge las teorías, marcos conceptuales o vocabularios que los propios estudios declaran. La segunda capa construye una lente analítica común para comparar estudios que no siempre usan las mismas palabras. La tercera capa fija límites interpretativos: qué puede afirmarse con los datos disponibles, qué aparece solo como señal emergente y qué no debe convertirse todavía en conclusión fuerte.

La revisión usa como lente analítica la relación amenaza-superficie-control-enforcement-resultado. La amenaza define la capacidad y el objetivo del atacante; la superficie localiza dónde entra o se materializa el riesgo; el control identifica el mecanismo que interrumpe la cadena; el enforcement determina cuándo una decisión deja de ser solo recomendación; y el resultado debe medir simultáneamente reducción de daño, utilidad, coste y fallo residual.

Modelo conceptual de la defensa operacional

En este marco, el **contrato comparativo** es la unidad mínima que hace interpretable una comparación: especifica amenaza y capacidad del atacante, superficie y activo protegidos, mecanismo de control, punto de enforcement, acción permitida o denegada, baseline, métricas de seguridad y utilidad, latencia, coste, robustez y fallo residual. No es una plantilla administrativa. Es la condición de conmensurabilidad que permite decidir si dos resultados prueban alternativas frente al mismo problema o si describen experimentos que solo comparten vocabulario.

El primer eje del modelo conceptual es la autoridad. Los ataques de prompt injection explotan la dificultad de distinguir instrucciones legítimas, datos no confiables y contenido recuperado que intenta adquirir autoridad. Los controles de procedencia, separación de contexto y flujo de información responden a esta ambigüedad; su hipótesis causal es que el sistema puede impedir que una fuente no autorizada determine una acción aunque el modelo procese su contenido.

El segundo eje es la capacidad. Un agente se vuelve peligroso cuando una salida textual puede transformarse en lectura de secretos, escritura, compra, envío, ejecución o persistencia. Autorización, mínimos privilegios, sandboxing y validación de llamadas actúan sobre esa transición. Su valor no depende de que el modelo reconozca siempre el ataque, sino de reducir qué puede hacer una decisión equivocada y de contener el radio de impacto.

El tercer eje es la observabilidad temporal. Los filtros de entrada actúan antes de que el modelo razone; los monitores de contexto y trayectoria actúan durante la construcción de la acción; los verificadores de salida actúan después de generar una respuesta; y los controles de memoria condicionan interacciones futuras.

Ningún punto observa por sí solo toda la cadena. La arquitectura defensiva debe situar el control antes del momento en que el daño se vuelve irreversible.

El cuarto eje es la evaluación. La seguridad empírica debe tratarse como una comparación adversarial y multiobjetivo, no como exactitud de clasificación aislada. ASR, falsos positivos, utilidad, latencia, coste, ataques no vistos, transferencia y disponibilidad describen dimensiones diferentes. La combinación de estas medidas determina si una defensa reduce riesgo operativo o solo desplaza el problema.

De estos ejes emerge una gramática de defensa operacional: autoridad, capacidad, observabilidad y evaluación. Esta gramática no presupone que una familia sea universalmente superior; permite explicar por qué ciertos controles pueden complementarse, por qué otros son redundantes y qué evidencia haría falta para sostener una frontera de dominancia entre configuraciones.

Los marcos teóricos explícitos aparecen de forma dispersa: Teoría de maleabilidad de memoria en pipeline write-retrieve-act; teorema de separación verificado por máquina (T1, T2, T3); control de flujo de información no maleable (non-malleable IFC) (n=1); Premisa: todo ataque IPI se manifiesta como una desviación de la trayectoria de acción esperada; no se formula una teoría formal (n=1); Geometría diferencial y teoría de local intrinsic dimensionality aplicadas al análisis de embeddings de lenguaje (n=1); Diagnóstico causal temporal para la toma de control en trayectorias multi-turno (n=1)

Metodológicamente, la síntesis focal contiene 50 estudios; 49 de ellos son empíricos. Dentro de los estudios empíricos, los diseños se distribuyen como Experimental: n=47, Cuantitativo: n=2. Esta distribución importa porque las conclusiones de una revisión sistemática no deben pesar igual cuando proceden de experimentos, benchmarks, estudios cualitativos, revisiones o propuestas teóricas.

Posicionamiento interpretativo del artículo

La aportación central del artículo es rechazar la idea de que un harness es mejor porque bloquea más ataques en un benchmark aislado. Una defensa solo puede considerarse superior dentro de un contrato operacional explícito: amenaza y atacante definidos, superficie protegida, punto de aplicación, baseline comparable, reducción de riesgo, utilidad preservada, coste asumible y fallo residual conocido.

Esta posición obliga a separar tres planos que muchas revisiones mezclan: lo que el corpus permite afirmar, lo que solo aparece como señal emergente y lo que todavía no puede sostenerse sin mejores diseños primarios.

En este artículo, mojarse no significa convertir frecuencias en certeza. Significa declarar una unidad de comparación, explicar por qué esa unidad ordena Harnesses de seguridad para modelos generativos y sistemas agénticos, reconocer qué evidencia la sostiene y dejar claro qué hallazgo futuro podría matizarla o refutarla.

Tesis analíticas del marco

1. La unidad de comparación no es el modelo ni el nombre del guardrail, sino el contrato operacional completo entre amenaza, superficie, control, punto de enforcement, baseline, eficacia, utilidad, coste y fallo residual.
2. Detección, prevención, contención y recuperación son funciones defensivas distintas. Un estudio debe declarar cuál implementa y qué daño puede ocurrir si esa función falla.
3. Cobertura estática y robustez adaptativa no son equivalentes. La segunda exige que el atacante pueda observar, reformular o trasladar su estrategia frente al control desplegado.
4. La superioridad es condicionada y multiobjetivo. Una defensa domina solo dentro de amenazas comparables y cuando mejora seguridad sin un deterioro desproporcionado de utilidad, disponibilidad o coste.
5. El modo de fallo no es una nota negativa, sino una propiedad teórica del harness: define la frontera donde hacen falta una capa compensatoria, recuperación o intervención humana.
6. La acumulación científica requiere taxonomías suficientemente precisas para no convertir controles no tipificados o benchmarks incompatibles en una mayoría artificial.

Método

El estudio se diseñó como una revisión sistemática de literatura sobre Harnesses de seguridad para modelos generativos y sistemas agénticos. La metodología combinó planificación explícita, búsqueda trazable, cribado documentado, lectura de texto completo y síntesis focal; los estándares de reporte de revisiones sistemáticas se usaron como soporte de transparencia, no como sustituto de la contribución analítica (Kitchenham & Charters, 2007; Snyder, 2019; Page et al., 2021; Rethlefsen et al., 2021).

Diseño de la revisión sistemática

La unidad de análisis fue el estudio publicado sobre Harnesses de seguridad para modelos generativos y sistemas agénticos. La ventana temporal fue entre enero de 2023 y agosto de 2026 y se cerró con fechas exactas en el protocolo y en `searches/search-log.csv`, evitando proyectar la revisión hasta un final de 2026 todavía no observado.

Antes de ejecutar la búsqueda se estableció un modo metodológico: Modo técnico. Esta decisión afecta al marco de pregunta, la descomposición de búsqueda, el cribado, la evaluación crítica, la ponderación del score focal y la forma de síntesis. En consecuencia, la revisión no aplica un molde único a cualquier disciplina: adapta la lógica de comparación a la unidad real de evidencia del campo.

No se registró un protocolo externo en PROSPERO ni en OSF antes de iniciar la revisión. Esta ausencia no se oculta ni se sustituye por una declaración retrospectiva: delimita el estatus del estudio. Como compensación de transparencia, el paquete conserva la captación inicial fechada, la pregunta, los criterios previos, la estrategia y descomposición de búsqueda, los logs por fuente, las decisiones de cribado, las discrepancias resueltas, los PDFs recuperados, la matriz de extracción y las reglas del corte focal. Una réplica puede así reconstruir qué decisiones precedieron a los resultados y cuáles surgieron durante la síntesis.

La revisión aplicó una regla conservadora de publicabilidad: DOI público normalizado, PDF local legible y extracción textual trazable. Esta regla puede excluir estudios potencialmente interesantes sin DOI o sin PDF recuperable, pero evita que el artículo final mezcle evidencia auditable con referencias imposibles de verificar.

El subconjunto focal no se decidió por intuición editorial. El protocolo fijó un rango entre 25 y 50 estudios para síntesis focal; después, solo los estudios incluidos tras lectura de texto completo fueron ordenados mediante un score compuesto transparente que se formaliza más abajo junto a la matriz de selección. Cuando el protocolo declara un rango, la regla no fuerza un número exacto: selecciona todos los estudios que cumplen DOI público, PDF legible, extracción trazable, confianza suficiente y score compuesto dentro del máximo autorizado, siempre que se alcance el mínimo metodológico. En esta revisión, el último estudio seleccionado fija un umbral efectivo de score $\geq 83,2$. Los estudios que cumplen la revisión pero no entran en ese corte se conservan como corpus contextual, de modo que la revisión no los borra ni los usa indebidamente como evidencia fina.

La puntuación fue diseñada como heurística transparente de priorización: la relevancia temática se calcula desde el ajuste a la pregunta, los criterios y las señales recuperables en título, resumen y texto completo; la calidad metodológica se apoya en diseño, muestra, instrumento, método, resultado y limitaciones recuperables; y la representatividad penaliza duplicación excesiva de una misma familia, fuente o tipo de tarea. El score focal no se presenta como una doble codificación humana: el manuscrito publica sus componentes, sensibilidad alternativa y trazabilidad DOI/PDF para que el corte pueda ser auditado externamente.

Como control de sensibilidad, el corte focal se recalculó con dos variantes de pesos del score: 0,40/0,40/0,20 y 0,45/0,30/0,25. En esta revisión, esas variantes conservaron 49/50 y 49/50 estudios del subconjunto focal. Esta prueba verifica que el N final no depende de una única ponderación oportunista, aunque no sustituye una codificación humana independiente del score.

Profundización metodológica y trazabilidad

La pregunta de investigación se tradujo en un protocolo operativo antes de interpretar resultados: tema, ventana temporal, fuentes, criterios, N objetivo, unidad de análisis y reglas de exclusión quedaron documentados como decisiones metodológicas previas. Esto es importante porque el método no debe aparecer solo como una narración posterior; debe funcionar como frontera previa que decide qué evidencia puede entrar, con qué peso y bajo qué condiciones.

La unidad de análisis fue el contrato operacional de defensa reportado por cada estudio sobre Harnesses de seguridad para modelos generativos y sistemas agénticos: activo protegido, amenaza, capacidad del atacante, superficie, arquitectura de control, punto de aplicación, decisión autorizada o denegada, baseline, métrica de seguridad, utilidad, latencia, coste, robustez y fallo residual. Esta decisión evita tratar como equivalentes un filtro de entrada, un monitor de trayectoria, una política de herramientas y un sandbox solo porque todos se presenten como guardrails.

La búsqueda se interpretó como captación reproducible y no como acumulación indiscriminada. Los 2440 registros identificados pasaron por deduplicación, cribado de título/resumen y recuperación de texto completo; los 2372 registros cribados no equivalen a estudios incluidos, sino al universo sometido a decisión inicial. El paso crítico fue la transición a texto completo: se buscaron 200 PDFs o textos completos, 57 no fueron recuperables y 143 pudieron evaluarse materialmente.

La decisión de cribado se organizó en tres estados: OK, KO o necesita más prueba. OK exige ajuste temático suficiente, DOI trazable y evidencia recuperable; KO exige motivo explícito de exclusión; necesita más prueba se usa cuando título y resumen no bastan para cerrar la decisión sin leer texto completo. Esta regla reduce el riesgo de excluir por intuición y obliga a que las decisiones dudosas avancen a una fase más exigente.

La lectura de texto completo funcionó como segunda frontera metodológica. Un registro no entró en el corpus final por promesa del abstract, popularidad del tema o afinidad con la tesis del artículo; entró cuando el PDF permitió recuperar evidencia suficiente sobre método, resultado y límite. Esta regla puede ser conservadora, pero protege la revisión frente a inferencias no auditables.

La extracción estructurada separó amenaza y superficie, control y punto de enforcement, atacante estático o adaptativo, corpus o entorno de evaluación, baseline, ASR o métrica defensiva equivalente, falsos positivos, utilidad, latencia, coste, transferencia, ablaciones, modos de fallo y disponibilidad de artefactos. Una cifra solo se codificó como resultado cuando el PDF permitía vincularla con el control y la evaluación observados; números de tablas, parámetros de diseño, coste del atacante o afirmaciones sin valor medido no se reutilizaron como rendimiento del harness.

La calidad no se trató como prestigio editorial, sino como extractabilidad y comparabilidad. El score combina relevancia temática, calidad metodológica y representatividad; además, la confianza de extracción valora si el PDF permite localizar lo que el artículo afirma. En el subconjunto focal, los vacíos de reporte quedan visibles: 32 estudios no declaran marco teórico suficiente, 18 estudios no detallan muestra, 20 estudios no detallan país o contexto y 5 estudios no explicitan variables o dimensiones analíticas.

La evaluación crítica se separó en dos capas. La primera capa valora calidad metodológica como parte del score de selección; la segunda capa, reportada en resultados, clasifica el riesgo de reporting y trazabilidad por estudio. Esta separación evita confundir prioridad de inclusión con certeza de inferencia: un estudio puede ser muy relevante para la pregunta y, aun así, dejar límites por comparador, teoría, validación externa o reporting incompleto.

La síntesis se dividió en dos niveles: corpus incluido y síntesis focal. Los 116 estudios incluidos delimitan el perímetro de la revisión; los 50 estudios focales sostienen la comparación intensiva. Esta separación evita dos errores frecuentes: presentar todo el corpus como si tuviera la misma densidad de evidencia o, al contrario, ocultar estudios válidos solo porque no entran en el N final.

El cribado de título y resumen conservó dos juicios automáticos independientes para 2372 registros. Coincidieron en 1868 casos y discreparon en 504; el acuerdo bruto fue 0,787 y el kappa de Cohen 0,490. Las discrepancias no se resolvieron mediante exclusión automática: se mantuvieron como **necesita más prueba**,

de modo que la incertidumbre favoreció elegibilidad para la siguiente frontera documental. En texto completo se conservaron 200 pares de juicio: 189 acuerdos y 11 discrepancias, con acuerdo bruto 0,945 y kappa de Cohen 0,888. 11 discrepancias quedaron resueltas mediante una decisión investigadora firmada después de leer el PDF y conservar las razones de ambos revisores y del adjudicador. El resultado final no descansa, por tanto, en que un modelo imponga silenciosamente su última respuesta. Estas métricas describen consistencia entre juicios automáticos independientes y no constituyen ground truth ni acuerdo interjueces humano. La extracción y la evaluación crítica tampoco se presentan como doble codificación humana; esa distinción se mantiene explícita para no confundir trazabilidad computacional con validación humana independiente.

La reproducibilidad se sostiene en artefactos concretos: protocolo, estrategia de búsqueda, logs por fuente, índice DOI, duplicados, registros cribados, decisiones de texto completo, matriz de extracción, matriz de score, anexos tabulares, PDFs locales cuando la licencia lo permite y paquete editorial. El método queda así conectado a archivos verificables y no solo a una declaración de buenas intenciones.

Criterios de inclusión

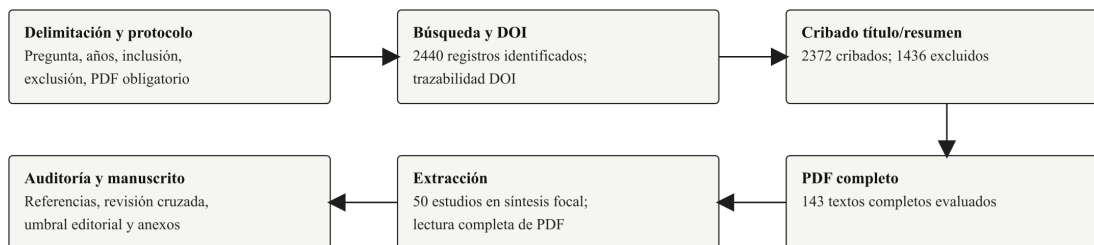
- Estudios con texto completo y una identidad bibliográfica trazable que implementen, describan con detalle evaluable o comparen controles de seguridad en tiempo de inferencia u operación para LLM, modelos multimodales o agentes; evaluaciones empíricas con amenaza, ataque, baseline o condición comparativa y al menos un resultado recuperable de seguridad, utilidad, coste, latencia, robustez o fallos positivos; trabajos de arquitectura que permitan reconstruir el punto de aplicación y la composición del harness; inglés o español.
- DOI público normalizado.
- PDF local legible para lectura de texto completo.

Criterios de exclusión

- Alineamiento o entrenamiento del modelo base sin capa de harness o control operacional; benchmarks de ataque sin defensa evaluada salvo uso contextual; ciberseguridad genérica sin relación directa con modelos generativos o agentes; opinión, marketing o propuesta sin método ni evidencia recuperable; duplicados; texto completo ilegible o no localizable; trabajos sin identidad bibliográfica pública y verificable.
- Material divulgativo, editorial, tutorial o meramente promocional sin contribución investigadora verificable.
- Mención tangencial del tema sin análisis, evaluación o resultado sustantivo.

Arquitectura operativa de revisión

Secuencia completa desde la delimitación de la pregunta hasta el manuscrito auditable.



Base metodológica: protocolo de revisión sistemática, PDF legible y cadena de auditoría editorial.

Figura 1: Arquitectura operativa de revisión.

Tabla 1. Flujo de selección de estudios.

Etapa	N
Registros identificados	2440
Duplicados consolidados antes del cribado	68
Cribado título/resumen	2372
Exclusiones título/resumen	1436
Candidatos tras título/resumen (include/maybe)	936
No trasladados a recuperación tras precheck documental	736
Candidatos enviados a recuperación de texto completo	200
PDF o texto completo no recuperado	57
PDF o texto completo recuperado y evaluado	143
Estudios incluidos tras lectura de texto completo	116
Estudios de síntesis focal	50
Estudios focales con DOI y PDF local legible	50

En la Tabla 1 no hay contradicción entre cribado y recuperación: tras título/resumen quedaron 936 registros include/maybe. De ellos, 736 no se trasladaron a recuperación por precheck documental, tipo de material, duplicidad funcional, falta de DOI utilizable o baja densidad metodológica antes de búsqueda de PDF; los 200 restantes sí pasaron a recuperación de texto completo. De esos 200, 57 no tuvieron PDF recuperable y 143 fueron evaluados a partir de PDF o texto completo extraído.

Tabla 2. Ejecución y cobertura operativa de la búsqueda por fuente.

Fuente	Eventos	Resultados brutos	Estado operacional
Crossref	28	2796	Ejecutada con resultados
OpenAlex	28	—	Omitida por cuota; reanudable con credencial
SemanticScholar	28	—	Interrumpida por error o límite de API
OpenAIRE	20	125	Ejecutada con resultados
arXiv	18	222	Ejecutada con resultados
Embase	1	—	Fuente opcional no ejecutada
IEEE Xplore	1	—	Fuente opcional no ejecutada
Lens	1	—	Fuente opcional no ejecutada
Scopus	1	—	Fuente opcional no ejecutada
Web of Science	1	—	Fuente opcional no ejecutada

Los resultados brutos de la Tabla 2 son respuestas acumuladas antes de deduplicación y pueden solaparse entre cadenas y fuentes. — no equivale a cero resultados: indica que la fuente no llegó a ejecutarse o que el log conserva una incidencia sin recuento recuperable. En particular, una única fila de Scopus, Web of Science, IEEE Xplore, Embase o Lens documenta su comprobación operacional y eventual omisión por acceso opcional, no una búsqueda equivalente a las APIs abiertas. Esta diferencia se conserva como límite de cobertura en lugar de presentar como ejecutada una fuente que no lo fue.

No produjeron un recuento recuperable OpenAlex, SemanticScholar, Embase, IEEE Xplore, Lens, Scopus y Web of Science. Esta carencia puede favorecer literatura accesible mediante APIs abiertas, repositorios y preprints frente a publicaciones indexadas solo en servicios institucionales. El sesgo no se corrige fingiendo equivalencia entre fuentes: se declara, se conserva en el log y limita la representatividad atribuible al corpus.

La estrategia de búsqueda combinó fuentes bibliográficas y APIs académicas. El log contiene estos eventos por fuente: OpenAlex: 28/127; Crossref: 28/127; SemanticScholar: 28/127; OpenAIRE: 20/127; arXiv: 18/127; Lens: 1/127; Scopus: 1/127; Web of Science: 1/127. Un evento registra tanto ejecuciones efectivas como omisiones justificadas, cuotas o errores recuperables; por eso la Tabla 2 separa intentos, resultados brutos y estado operacional. Las consultas se ejecutaron el 2026-08-02, con ventana registrada entre 2023-01-01 y 2026-08-02. La lógica de consulta se construyó alrededor del tema

Harnesses de seguridad para modelos generativos y sistemas agénticos mediante bloques de concepto, sinónimos y combinaciones bilingües cuando el campo lo requería. Ejemplos de cadenas de búsqueda: "LLM security guardrails" evaluation; "large language model" "runtime guardrail" security; "AI agent" "security harness"; "LLM firewall" evaluation; "policy enforcement" "LLM agent"; "defense in depth" "large language model"; y 73 cadenas adicionales en el anexo. El archivo `searches/search-log.csv` conserva para cada consulta la fuente, plataforma, cadena exacta, fecha de consulta, ventana temporal, notas y fichero exportado; `protocol/search-strategy.md` y `protocol/search-decomposition.md` documentan la descomposición de la pregunta en estadios de búsqueda.

Tabla 3. Reglas operativas de composición del subconjunto focal.

Criterio	Regla operativa
Elegibilidad de entrada	solo estudios incluidos tras lectura de texto completo
DOI público	sin DOI normalizado no entra en el corpus publicable
PDF local legible	sin texto completo extraído no entra en síntesis focal
Relevancia temática	50% del score compuesto; mide ajuste a pregunta y criterios
Calidad metodológica	35% del score compuesto; mide diseño, método, muestra, evaluación y trazabilidad
Representatividad	15% del score compuesto; evita concentración por fuente o familia casi idéntica
Corte focal	entran los estudios robustos dentro del contrato un rango entre 25 y 50 estudios; en esta revisión el umbral efectivo del score compuesto fue $\geq 83,2$
Umbral de confianza	la síntesis focal exige extracción robusta; por defecto se usa confianza ≥ 80 salvo que el protocolo documente una reserva excepcional
Sensibilidad	solapamiento alternativo: 49/50 y 49/50

Tabla 4. Rúbrica operativa de confianza de extracción.

Indicador	Peso máximo
Ajuste explícito a la pregunta de investigación	25
Claridad de diseño, muestra o unidad de análisis	20
Método, instrumento, métrica o procedimiento reportado	20
Resultado sustantivo y limitaciones recuperables	20
Trazabilidad dentro del PDF y consistencia de extracción	15

Tabla 4A. Componentes auditables del score en estudios focales.

Pos.	DOI	Rel.	Calidad	Rep.	Score
1	10.48550/arxiv.2502.05174	100,0	95,0	50,0	90,8
2	10.48550/arxiv.2501.18837	96,0	92,0	50,0	87,7
3	10.48550/arxiv.2606.24322	95,0	92,0	50,0	87,2
4	10.48550/arxiv.2512.06716	95,0	90,0	50,0	86,5
5	10.48550/arxiv.2503.03502	95,0	90,0	50,0	86,5
6	10.48550/arxiv.2511.15203	95,0	90,0	50,0	86,5
7	10.48550/arxiv.2602.22724	95,0	90,0	50,0	86,5

Pos.	DOI	Rel.	Calidad	Rep.	Score
8	10.48550/arxiv.2605.17986	95,0	90,0	50,0	86,5
9	10.48550/arxiv.2506.01055	95,0	90,0	50,0	86,5
10	10.48550/arxiv.2606.02240	95,0	90,0	50,0	86,5
11	10.48550/arxiv.2605.11039	95,0	90,0	50,0	86,5
12	10.48550/arxiv.2605.01970	95,0	90,0	50,0	86,5
13	10.48550/arxiv.2604.24118	95,0	88,0	50,0	85,8
14	10.48550/arxiv.2605.26497	95,0	88,0	50,0	85,8
15	10.48550/arxiv.2603.21642	95,0	88,0	50,0	85,8
16	10.48550/arxiv.2603.10749	95,0	88,0	50,0	85,8
17	10.48550/arxiv.2601.10173	95,0	88,0	50,0	85,8
18	10.48550/arxiv.2508.10991	95,0	88,0	50,0	85,8
19	10.48550/arxiv.2601.12449	95,0	88,0	50,0	85,8
20	10.48550/arxiv.2601.04603	95,0	88,0	50,0	85,8
21	10.48550/arxiv.2605.10582	95,0	88,0	50,0	85,8
22	10.48550/arxiv.2605.03482	95,0	88,0	50,0	85,8
23	10.48550/arxiv.2606.17467	95,0	88,0	50,0	85,8
24	10.48550/arxiv.2605.03095	95,0	88,0	50,0	85,8
25	10.48550/arxiv.2607.13081	95,0	88,0	50,0	85,8
26	10.48550/arxiv.2606.20922	95,0	88,0	50,0	85,8
27	10.21203/rs.3.rs-9932271/v1	95,0	88,0	50,0	85,8
28	10.48550/arxiv.2508.09288	96,0	86,0	50,0	85,6
29	10.48550/arxiv.2510.19169	95,0	85,0	50,0	84,8
30	10.48550/arxiv.2604.11790	95,0	85,0	50,0	84,8
31	10.48550/arxiv.2604.07223	95,0	85,0	50,0	84,8
32	10.48550/arxiv.2507.15219	95,0	85,0	50,0	84,8
33	10.48550/arxiv.2512.06556	95,0	85,0	50,0	84,8
34	10.48550/arxiv.2602.05066	95,0	85,0	50,0	84,8
35	10.48550/arxiv.2606.15441	95,0	85,0	50,0	84,8
36	10.48550/arxiv.2601.04795	95,0	85,0	50,0	84,8
37	10.48550/arxiv.2604.23887	95,0	85,0	50,0	84,8
38	10.48550/arxiv.2602.20708	95,0	85,0	50,0	84,8
39	10.48550/arxiv.2601.03300	95,0	85,0	50,0	84,8
40	10.48550/arxiv.2604.03870	95,0	85,0	50,0	84,8
41	10.1609/aaai.v40i44.41074	95,0	85,0	50,0	84,8
42	10.21203/rs.3.rs-10196595/v1	95,0	85,0	50,0	84,8
43	10.48550/arxiv.2512.00966	95,0	84,0	50,0	84,4
44	10.48550/arxiv.2510.05244	95,0	82,0	50,0	83,7
45	10.48550/arxiv.2605.02812	95,0	82,0	50,0	83,7
46	10.48550/arxiv.2602.10915	95,0	82,0	50,0	83,7
47	10.48550/arxiv.2603.07191	95,0	82,0	50,0	83,7
48	10.48550/arxiv.2603.28807	95,0	82,0	50,0	83,7
49	10.48550/arxiv.2410.22770	92,0	85,0	50,0	83,2
50	10.48550/arxiv.2511.15759	92,0	85,0	50,0	83,2

Nota metodológica sobre la relevancia: cuando todos los estudios focales muestran una relevancia similar, la columna **Rel.** debe leerse como puerta de elegibilidad temática, no como una jerarquía fina. En ese caso, la discriminación real del subconjunto focal procede de calidad metodológica recuperable, representatividad/diversidad, confianza de extracción, disponibilidad de PDF y estabilidad del corte. La tabla conserva **Rel.** precisamente para mostrar que no se introducen décimas artificiales de relevancia cuando el corpus ya pasó un umbral temático homogéneo.

Tabla 4B. Estatus de estudios contextuales elegibles pero no focales.

Pos./rango	N o DOI	Score	Rel.	Calidad	Rep.	Criterio operativo
51-116	66	52,5-83,2	70,0-95,0	35,0-85,0	35,0-55,0	Perímetro contextual elegible; fuera del corte focal por score inferior al umbral efectivo o menor densidad comparativa

La Tabla 4B expone score o rango de score para que el recorte focal no quede como decisión opaca. Estos registros pertenecen al perímetro contextual elegible de la revisión, pero no alcanzaron el umbral combinado que justifica tratarlos al mismo nivel que el N focal. La distinción metodológica relevante es focal frente a contextual: reordenar internamente ese perímetro exigiría una segunda ronda de extracción profunda o una validación humana adicional.

Tabla 4C. Comparación focal-contextual.

Grupo	N	Perfil dominante	Vacíos de reporte	Score/estatus	Fuentes principales
Síntesis focal	50	Empírico; Experimental	sin contexto: 20; sin muestra: 28	85,5	arxiv: 29/50; openaire: 18/50; crossref: 3/50
Perímetro contextual elegible no focal	66	Otros; sin diseño empírico comparable	sin contexto: 66; sin muestra: 66	75,0	arxiv: 34/66; crossref: 17/66; openaire: 15/66

La Tabla 4C verifica que el recorte focal no se trate como una caja negra: compara el núcleo intensivo con los estudios incluidos que permanecen como contexto, mostrando perfil dominante, vacíos de reporte, score medio y fuentes. Si ambos grupos difieren, esa diferencia se interpreta como límite de transferencia de la síntesis focal y no como desaparición metodológica del corpus contextual. En el perímetro contextual, un campo vacío significa que esa variable no fue recuperada en la matriz compacta de selección; no demuestra por sí solo que el paper original no la reporte ni que el estudio sea de menor calidad. El rasgo común de ese perímetro es una densidad comparativa o un score inferiores al umbral focal. Elevar uno de esos registros al mismo plano analítico exige lectura y extracción profunda adicionales, no una inferencia desde el vacío tabular.

La puntuación base de la preselección focal se calculó mediante una regla ponderada explícita.

La regla de priorización focal puede expresarse formalmente como:

$$\text{Score}_i = 0,50 \text{ Rel}_i + 0,35 \text{ Cal}_i + 0,15 \text{ Rep}_i$$

donde Rel_i representa la relevancia temática del estudio, Cal_i su calidad metodológica recuperable y Rep_i la corrección de representatividad/diversidad aplicada antes del corte focal. Las ponderaciones proceden del Modo técnico y las tres magnitudes se expresan en escala 0-100.

Operativamente, la relevancia se asigna por bandas de ajuste temático: sin ajuste suficiente, ajuste tangencial, ajuste directo a la pregunta y ajuste directo con mecanismo, variables y contexto recuperables. La calidad combina confianza de extracción, claridad de diseño, muestra/unidad, método, comparador, resultado y limitaciones recuperables. La representatividad aplica una corrección de diversidad para no saturar el N focal con la misma fuente, familia temática, país, plataforma, diseño o registro casi duplicado. Los valores crudos y el score resultante se conservan en `paper/appendices/data/selection-score-matrix.csv`.

La representatividad no se interpreta como calidad adicional del artículo, sino como corrección de diversidad: evita que el N focal quede saturado por una misma fuente, familia temática, tipo de trabajo, país, plataforma, benchmark o propuesta casi duplicada. En empates o bandas equivalentes, se conserva el orden estable de la matriz auditada y no se reordenan manualmente los registros después de conocer los resultados.

Tabla 4D. Operacionalización auditable de los componentes del score.

Componente	Escala	Criterios observables
Relevancia temática (Rel)	0-100	Ajuste a la pregunta, población/contexto, tecnología o intervención, tarea/dominio, ventana temporal y criterios de inclusión.
Calidad metodológica recuperable (Cal)	0-100	Claridad de diseño, muestra o unidad analítica, método/instrumento, comparador, resultado, limitaciones y trazabilidad textual en PDF.
Representatividad/diversidad (Rep)	0-100	Corrección para evitar que el N focal quede dominado por una fuente, familia de tarea, tipo de estudio, país, año o propuesta casi duplicada.
Desempate y estabilidad	orden estable	Cuando varios estudios quedan en la misma banda de score, no se introduce una jerarquía artificial: se conserva el orden auditado, se reporta el umbral y se deja una matriz suplementaria para reproducir o reabrir el corte.
Corte operativo	score compuesto	El ranking se aplica solo a estudios ya incluidos por full text; no sustituye criterios de elegibilidad ni elimina del corpus contextual los estudios válidos fuera del N focal.

La Tabla 4A identifica mediante DOI los 50 estudios que sostienen la síntesis intensiva y desglosa relevancia, calidad, representatividad y score para cada uno; la Tabla 4B muestra score o rango de score de los registros contextuales; y la matriz suplementaria conserva títulos, DOI y razones operativas para auditar la selección del N final sin depender de una decisión opaca del manuscrito. El umbral $\geq 83,2$ debe leerse como resultado operativo de la regla aplicada al corpus focal, no como una frontera inventada después de leer los resultados. Las variantes de sensibilidad mantienen el solapamiento indicado en la Tabla 3 y funcionan como control contra un corte oportunista. ## Análisis estructural del corpus

Como análisis complementario, se construyeron redes separadas de coautoría, citación, acoplamiento bibliográfico, cocitación, coocurrencia temática y relaciones entre estudios y dimensiones extraídas. La unidad de identidad de estudio fue el DOI normalizado. Se calcularon grado ponderado, intermediación, centralidad armónica, PageRank, centralidad de autovector, núcleo k, clustering y participación entre comunidades. La intermediación ponderada convirtió la fuerza de vínculo en distancia mediante $d_{ij} = 1/w_{ij}$.

Las comunidades se estimaron con Louvain para múltiples semillas y resoluciones; la partición de mayor modularidad se contrastó mediante información mutua normalizada. Su interpretación exigió al menos 20

estudios incluidos, estabilidad igual o superior a 0,80 y cobertura suficiente de la capa correspondiente. Sobre un denominador de 50 estudios, la cobertura fue 100.0% para autoría, 0.0% para referencias y 42.0% para palabras clave. La productividad, las citas y la posición de red no intervinieron en la elegibilidad, la evaluación crítica ni la selección focal. Los parámetros y denominadores se conservan en los anexos estructurales.

Resultados

Los resultados no se presentan como una contabilidad plana de estudios sobre Harnesses de seguridad para modelos generativos y sistemas agénticos. La decisión científica que habilitan es identificar qué unidad de comparación permite acumular evidencia sin borrar diferencias de diseño, medición, contexto y calidad.

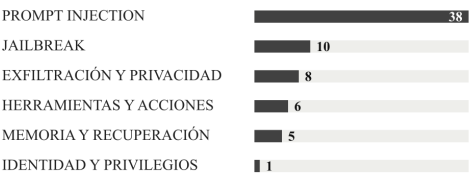
El aporte al campo es una regla de decisión que sustituye rankings universales por fronteras de dominancia. Un harness domina a otro solo si reduce más riesgo bajo la misma amenaza y baseline sin empeorar de manera material utilidad, falsos positivos, coste o latencia; cuando existen compensaciones, la conclusión correcta es contextual y debe nombrar qué propiedad se prioriza. En consecuencia, las tablas no funcionan como cierre descriptivo: funcionan como soporte para decidir qué afirmaciones son defendibles, cuáles son solo señales y qué vacíos impiden una conclusión más fuerte (Huang et al., 2026; L. Zhao et al., 2026; Yung et al., 2025; Z. Lin et al., 2026).

El mapa documental conserva la trazabilidad completa: 116 estudios incluidos, 50 estudios en síntesis focal y 66 estudios como perímetro contextual elegible. A partir de esos 116 estudios incluidos, se definió una síntesis focal de 50 estudios para la comparación intensiva. Los 66 estudios restantes siguieron dentro del corpus de revisión como perímetro contextual elegible: delimitan el campo y conservan trazabilidad, pero no alcanzaron el umbral combinado de ajuste, calidad, representatividad y densidad extractiva que exige la comparación intensiva. Por eso se reportan como contexto auditable y no como evidencia fina del N focal.

Panorama de amenazas y controles

Distribución de amenazas, superficies y familias de control evaluadas en el corpus final.

(A) AMENAZAS EVALUADAS



(B) FAMILIAS DE CONTROL



(C) LECTURA COMPARATIVA

Mayor presencia: Prompt injection (n=38) y Monitor de ejecución (n=12).
Control no tipificado n=10: esta incertidumbre limita cualquier ranking por familia.
Los conteos son multietiqueta: un estudio puede contribuir a varias familias.

Base: n=50 estudios focales. Clasificación multietiqueta derivada de taxonomía amenaza-control normalizada desde texto completo.

Figura 2: Panorama temático del corpus final.

La Figura 2 compara, en el panel izquierdo, la cobertura de amenazas y superficies y, en el panel derecho, las familias de control. Las barras representan conteos multietiqueta, por lo que un estudio puede contribuir a varias categorías. La concentración en prompt injection y monitores de ejecución describe dónde existe más investigación, no qué control es más eficaz; la categoría no tipificada hace visible la parte que todavía no admite ranking por mecanismo. **Familias de control** nombra aquí el conjunto amplio de funciones del harness; **guardrail** se reserva para controles de clasificación, filtrado o política y no se usa como sinónimo del sistema completo. El patrón principal es que la eficacia defensiva no forma una jerarquía única: depende

de amenaza, superficie, punto de aplicación, atacante, baseline y compensación entre seguridad, utilidad y coste.

Matriz amenaza-control

Cobertura observada entre amenazas y controles; una celda vacía significa ausencia de evidencia recuperable, no eficacia nula.

CRUCE ENTRE ARQUITECTURA DE CONTROL Y AMENAZA EVALUADA

	PROMPT INJECTION	JAILBREAK	EXFILTRACIÓN Y PRIVACIDAD	HERRAMIENTAS Y ACCIONES	MEMORIA Y RECUPERACIÓN	IDENTIDAD Y PRIVILEGIOS
MONITOR DE EJECUCIÓN	8	4	2	2		1
CONTROL DE MEMORIA	5		3		4	1
CONTROL NO TIPIFICADO	9	1	1	2		
GUARDRAIL DE POLÍTICAS	8	1	3	1		
VERIFICACIÓN CAUSAL	5	1	1	1		
PROCEDENCIA Y FLUJO	5	1			1	

LECTURA COMPARATIVA

- Cruce tipificado con mayor cobertura: Monitor de ejecución x Prompt injection (n=8).
- Control no tipificado en 13 cruces visibles; requiere lectura antes de comparar.

Matriz multietiqueta: un estudio puede evaluar varias amenazas y controles; una celda vacía no implica eficacia nula.

Figura 3: Matriz entre temas, tareas, métodos y resultados observados.

La Figura 3 cruza amenazas en columnas y familias de control en filas. La intensidad y el número de cada celda indican cuántos estudios focales evalúan ese cruce; una celda vacía significa ausencia de evidencia recuperable, no eficacia nula. El mapa muestra dónde pueden buscarse replicaciones comparables y dónde la literatura aún carece de cobertura entre superficie y control.

Mapa de comparabilidad metodológica

Diseños de evaluación y vacíos de reporting que limitan la comparación entre harnesses de seguridad.

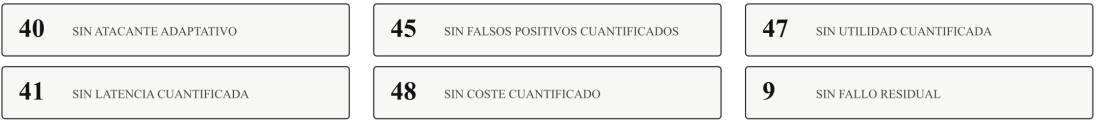
(A) DISEÑOS EMPÍRICOS (n=49)



(B) CRITERIOS REPORTADOS (base n=49)



(C) VACÍOS DE REPORTE QUE LIMITAN LA COMPARACIÓN



Base: 49 estudios con lectura metodológica; diseños empíricos n=49.

Figura 4: Mapa de comparabilidad metodológica de los estudios empíricos.

La Figura 4 resume el nivel de comparabilidad metodológica del subconjunto focal: diseño, muestra, contexto, variables, comparador y validación.

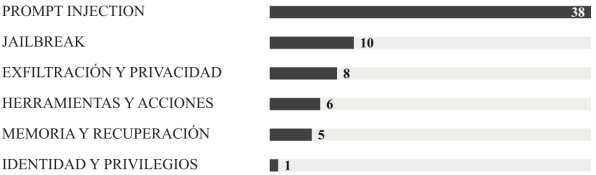
Madurez comparativa de la evidencia

Qué harnesses admiten comparación condicional y dónde faltan piezas del contrato experimental.

(A) MADUREZ DEL CONTRATO COMPARATIVO



(B) AMENAZAS CON MAYOR COBERTURA



(C) REGLA DE DOMINANCIA

Un harness solo entra en una frontera comparable si comparte amenaza, baseline, efecto de seguridad, coste operativo y prueba de robustez. Sin ese contrato no se infiere superioridad.

Base: 50 estudios focales y 37 configuraciones amenaza-control identificadas.

Figura 5: Madurez comparativa de la evidencia por estado y dominio de resultado.

La Figura 5 separa alineación descriptiva, evidencia insuficiente y preguntas abiertas. Esta distinción evita presentar como consenso causal lo que todavía es repetición documental, heterogeneidad no resuelta o una comparación aislada.

Convergencias, desacuerdos y preguntas abiertas

La síntesis no convierte la repetición de términos en consenso. Cada hallazgo se relacionó con una unidad de comparación y se clasificó de forma conservadora como asociación positiva, asociación negativa, resultado nulo, patrón condicionado, aportación descriptiva o dirección no recuperable. Esta operación permite separar coincidencia sustantiva de similitud verbal.

El mapa identifica 0 unidades con convergencia directa, 1 con alineación entre contextos, 3 con alineación descriptiva, 0 con desacuerdo o inconsistencia y 35 que permanecen abiertas o con evidencia insuficiente. Estos conteos no se interpretan como efectos agregados: dos estudios solo ocupan el mismo plano cuando comparten constructo o exposición, resultado, unidad de análisis y contexto suficientemente comparables.

Por tanto, un valor cero en convergencia directa describe el resultado de un umbral exigente de commensurabilidad, no una propiedad esencial del fenómeno ni una afirmación de que ningún estudio obtenga resultados compatibles. La compatibilidad bajo contratos parciales se conserva como alineación entre contextos, alineación descriptiva o pregunta abierta.

Un valor cero en desacuerdo no significa que el campo carezca de resultados opuestos. Significa que no se identificaron desacuerdos entre estudios que conservaran una unidad de comparación suficientemente equivalente. Los resultados incompatibles bajo otra amenaza, métrica, modelo, contexto o baseline permanecen como heterogeneidad o pregunta abierta y no se fuerzan dentro de una falsa contradicción directa.

Tabla de posiciones de evidencia. Unidades compartidas por al menos dos estudios.

Unidad de comparación	N	Estado	Direcciones observadas
LLM → Éxito del ataque	7	Alineación descriptiva	Aportación descriptiva o teórica, Dirección no recuperable
LLM → Éxito del ataque + Falsos positivos	3	Alineación descriptiva	Dirección no recuperable

Unidad de comparación	N	Estado	Direcciones observadas
Presencia ausencia defensa → Éxito del ataque	2	Alineación entre contextos	Dirección no recuperable, Asociación positiva
LLM → Eficiencia y recursos	2	Evidencia insuficiente	Dirección no recuperable, Patrón mixto o condicionado
LLM → Eficiencia y recursos + Calidad y rendimiento	2	Alineación descriptiva	Dirección no recuperable

La dirección estadística y la utilidad práctica se conservaron por separado. Una reducción de latencia, error, coste o riesgo puede ser una relación negativa y, al mismo tiempo, un resultado favorable; del mismo modo, un aumento puede ser adverso cuando crece una carga o un riesgo. Esta distinción evita traducir automáticamente «positivo» como beneficioso y «negativo» como perjudicial.

Dirección no recuperable indica que el texto completo no permite reconstruir el signo o patrón de la relación con suficiente trazabilidad. **No determinada**, en cambio, se refiere a la lectura práctica: puede existir un resultado reportado, pero la evidencia no permite clasificarlo de forma estable como favorable, adverso o dependiente del contexto.

Tabla de dominios de resultado. Señales transversales sin agregación causal.

Dominio de resultado	N	Tipo de señal	Lectura práctica
Éxito del ataque	36	Señal transversal	Adversa, Contextual, Favorable, No aplicable, Compensación o resultado mixto, No determinada
Eficiencia y recursos	12	Señal transversal	Favorable, Compensación o resultado mixto, No determinada
Calidad y rendimiento	11	Señal transversal	Favorable, Dependiente del contexto, Compensación o resultado mixto, No determinada
Defensa y detección	5	Señal transversal	Adversa, Favorable
Falsos positivos	5	Señal transversal	Favorable, No determinada
Seguridad y riesgo	2	Señal transversal	Dependiente del contexto, No determinada
Utilidad preservada	2	Señal transversal	Favorable, No determinada

La matriz completa conserva DOI, diseño, contexto, fragmento y localización. Por ello, una discrepancia no se resuelve por mayoría documental: se examina si procede de una medición distinta, un moderador, otra población, un diseño menos fuerte o una diferencia real entre resultados.

Tabla 5. Distribución del corpus incluido por estado analítico.

Estado analítico	N
Estudios empíricos dentro de la síntesis focal	49 (42,2%)
Estudios teóricos, de revisión o metodológicos dentro de la síntesis focal	1 (0,9%)
Perímetro contextual elegible no focal	66 (56,9%)

En la Tabla 5, el estado analítico separa el corpus incluido completo de la síntesis focal. La revisión contiene 116 estudios incluidos: 50 sostienen la comparación intensiva y 66 permanecen como perímetro contextual elegible no focal. Dentro de la síntesis focal, 49 estudios son empíricos y 1 aportan soporte teórico, metodológico o de revisión. Esta separación evita mezclar tipo de trabajo con función analítica dentro del manuscrito; los porcentajes se calculan sobre el N incluido total para mostrar la arquitectura completa del corpus.

Tabla 5A. Función probatoria de los estudios focales según tipo de evidencia.

Capa de evidencia	N	Función dentro de la síntesis	Peso interpretativo
Evidencia empírica directa	49	Sostiene la respuesta principal a la pregunta de investigación mediante datos, muestra, corpus, experimento, encuesta, panel, caso o evaluación observable.	Base probatoria para afirmaciones sustantivas.
Soporte teórico, metodológico o de revisión	1	Aporta vocabulario conceptual, mecanismos plausibles, límites de medición, contexto disciplinar o contraste con revisiones previas.	Apoyo interpretativo; no se contabiliza como efecto empírico equivalente.

Esta separación es importante porque la pregunta de investigación pide evidencia empírica. Por ello, los 49 estudios empíricos sostienen el peso probatorio de la respuesta, mientras que los 1 trabajos no empíricos cumplen una función auxiliar: ordenar conceptos, mecanismos, límites de medición y condiciones de interpretación. La síntesis no los usa para inflar la evidencia empírica ni para cerrar efectos causales; los usa para explicar por qué ciertos hallazgos pueden compararse y por qué otros deben mantenerse como contexto. La Tabla 5A clasifica función probatoria; la Tabla 7, en cambio, clasifica familia sustantiva dominante. Por eso sus N no deben sumarse ni compararse como si midieran lo mismo: una tabla responde qué peso tiene cada tipo de evidencia y la otra responde qué tema organiza cada estudio focal.

Tabla 6. Distribución del subconjunto empírico por diseño.

Tipo empírico	N
Experimental	47 (95,9%)
Cuantitativo	2 (4,1%)

Nota: los porcentajes de las tablas descriptivas se redondean a una decimal y pueden no sumar exactamente 100%.

Tabla 7. Síntesis compacta del subconjunto focal por familia de evidencia.

Familia	N	Diseño dominante	Lectura sintética
Herramientas, permisos y aislamiento	29	Experimental	29 estudios sitúan la seguridad en la capacidad de actuar: permisos mínimos, sandbox y validación de llamadas reducen impacto aunque el modelo sea manipulado.
Contexto, RAG y prompt injection indirecta	8	Experimental	8 estudios tratan documentos, recuperación y memoria como entradas no confiables y desplazan el control hacia procedencia, aislamiento y separación de instrucciones.
Jailbreak y cumplimiento de políticas	8	Experimental	8 estudios miden resistencia a evasión de políticas, pero su fuerza depende de incluir ataques adaptativos y utilidad legítima.
Exfiltración, secretos y privacidad	4	Experimental	4 estudios evalúan si el control evita que datos o secretos crucen límites, una propiedad que no queda cubierta por bloquear lenguaje ofensivo.
Monitorización, verificación y salida	1	Experimental	1 estudios colocan detección y decisión durante o después de la inferencia, con el riesgo de reaccionar tarde si ya ocurrió una acción.

Las filas de la Tabla 7 cubren el subconjunto focal completo: 50 de 50 estudios. La diferencia entre 50 estudios focales y 49 estudios empíricos se debe a que el subconjunto puede contener revisiones, trabajos teóricos o estudios metodológicos que aportan estructura conceptual pero no se clasifican como empíricos.

Tabla 8. Señales agregadas que sostienen la síntesis principal.

Dimensión	Señal agregada	Lectura para la revisión
Superficies y familias defensivas	Herramientas, permisos y aislamiento: 29/50; Jailbreak y cumplimiento de políticas: 8/50; Contexto, RAG y prompt injection indirecta: 8/50; Exfiltración, secretos y privacidad: 4/50; Monitorización, verificación y salida: 1/50	La clasificación separa la superficie protegida; un mismo estudio puede combinar capas, pero se resume por su función defensiva dominante.
Contrato mínimo de comparación	amenaza: 50/50; control: 50/50; punto de aplicación: 50/50; baseline: 45/50	Sin amenaza, control, enforcement y baseline recuperables no puede sostenerse una comparación de superioridad.
Eficacia defensiva	ASR o métrica equivalente: 30/50; ataques adaptativos: 15/50; robustez: 49/50	La cobertura de ataques estáticos informa rendimiento local; la robustez exige adaptación, transferencia o ataques no vistos.
Coste de seguridad	falsos positivos: 13/50; utilidad: 31/50; latencia: 9/50; coste: 4/50	Un harness no es mejor si reduce ataques a costa de bloquear uso legítimo o introducir un sobre coste operacional no declarado.
Fallo residual y reproducibilidad	modos de fallo: 41/50; código o artefacto: 38/50	Reportar fallos y artefactos permite distinguir una defensa generalizable de un ajuste opaco al benchmark.

Tabla 8A. Fronteras de dominancia condicionada entre familias de harnesses.

Amenaza	Familia de control	N	Baseline	Atacante adaptativo	Trade-offs cuantificados (N)	Estado
Prompt injection	Procedencia y flujo de información + Autorización de herramientas	2	2/2	1/2	FP 1; utilidad 2; latencia/coste 2	Candidata replicada
Prompt injection	Filtrado de salida	3	3/3	2/3	FP 0; utilidad 1; latencia/coste 0	Frontera emergente
Prompt injection	Verificación causal o contrafactual	2	1/2	0/2	FP 0; utilidad 1; latencia/coste 1	Frontera emergente
Prompt injection	Procedencia y flujo de información + Control de memoria o recuperación	2	2/2	0/2	FP 0; utilidad 1; latencia/coste 0	Frontera emergente
Envenenamiento de memoria o RAG + Exfiltración de datos	Control de memoria o recuperación	2	2/2	1/2	FP 0; utilidad 0; latencia/coste 0	Eficacia sin coste completo
Prompt injection	Guardrail de política o intención	2	1/2	1/2	FP 1; utilidad 0; latencia/coste 0	Eficacia sin coste completo
Prompt injection	Control no tipificado de forma homogénea	6	6/6	4/6	FP 1; utilidad 2; latencia/coste 2	No tipificable
Prompt injection + Abuso o envenenamiento de herramientas	Control no tipificado de forma homogénea	2	2/2	0/2	FP 0; utilidad 1; latencia/coste 1	No tipificable

Los estados se asignan con reglas operativas distintas. **Candidata replicada** exige al menos dos estudios de la misma combinación tipificada de amenaza y control con evidencia suficiente para probar dominancia condicionada. **Frontera emergente** indica que existe una comparación con trade-offs recuperables, pero falta réplica o robustez. **Eficacia sin coste completo** identifica reducción de ataque sin cobertura suficiente de utilidad, falsos positivos, latencia o coste. **Comparación insuficiente** señala que ni eficacia ni compensaciones permiten una comparación justa. **No tipificable** se reserva para grupos residuales donde la amenaza o el mecanismo de control no pueden clasificarse de forma homogénea; disponer de baseline no corrige esa falta de equivalencia mecánica.

En las columnas **Baseline** y **Atacante adaptativo**, la fracción x/y significa número de estudios que reportan la característica sobre el total de estudios de esa familia. En **Trade-offs**, FP indica falsos positivos y cada recuento identifica cuántos estudios cuantifican utilidad o latencia/coste; una fracción alta no expresa mayor eficacia, sino mejor cobertura de reporte.

La señal comparativa más madura aparece en Prompt injection con Procedencia y flujo de información + Autorización de herramientas (n=2). Debe leerse como dirección prioritaria para evaluación y diseño, no como ganador universal: la equivalencia de amenaza, baseline, métrica y coste todavía debe comprobarse estudio por estudio.

Además, 10 configuraciones quedan en familias residuales no tipificadas. Su volumen no se interpreta como réplica: agrupar mecanismos desconocidos produciría una falsa mayoría.

La no tipificación aparece cuando el estudio describe el control con una etiqueta genérica, combina mecanismos sin aislar su contribución o no aporta detalle operacional suficiente para mapearlo de forma estable a procedencia, autorización, aislamiento, memoria, verificación o filtrado. La categoría conserva esos trabajos sin atribuirles una homogeneidad defensiva que la fuente no permite sostener.

La tabla no agrega ASR, falsos positivos o utilidad entre benchmarks incompatibles. Su función es identificar dónde existe evidencia suficiente para una comparación condicionada y dónde el siguiente estudio debería replicar amenaza, baseline y costes antes de afirmar superioridad.

Lectura interpretativa de los resultados

Antes de continuar con la síntesis, el artículo fija qué lectura autoriza el corpus y qué lectura no autoriza. Esta matriz evita que las tablas funcionen como decoración: cada número debe convertirse en una afirmación, una cautela o una condición de frontera.

Tabla 8B. Matriz de toma de posición interpretativa.

Plano	Lectura del artículo	Base o cautela
Afirmación que sí sostiene el artículo	La superioridad defensiva solo puede afirmarse dentro de un contrato compartido de amenaza, superficie, atacante, punto de aplicación y baseline.	Amenaza recuperable en 50/50 estudios, baseline en 45/50 y eficacia defensiva cuantificada en 30/50.
Afirmación que no sostiene	No existe evidencia para ordenar todos los harnesses en un ranking universal ni para tratar una tasa de bloqueo aislada como seguridad operacional.	Solo 15/50 estudios prueban atacantes adaptativos y la cobertura de utilidad, falsos positivos, latencia y coste sigue siendo desigual.
Condición de frontera	Una configuración domina de forma condicionada cuando reduce el riesgo frente al mismo atacante y baseline sin degradar de manera material la utilidad o desplazar un coste inaceptable.	Si seguridad, utilidad y coste se compensan, el resultado correcto es una frontera de decisión, no un campeón único.

Plano	Lectura del artículo	Base o cautela
Qué cambiaría la tesis	Replicaciones entre modelos, dominios y ataques adaptativos, con artefactos abiertos y medición conjunta de seguridad, utilidad, latencia, coste y fallo residual.	Robustez o transferencia aparece en 49/50 estudios y modos de fallo en 41/50; ampliar ambas señales permitiría convertir fronteras emergentes en recomendaciones más fuertes.

Síntesis sustantiva de resultados

La respuesta sustantiva no es que exista un harness universalmente mejor. Los 50 estudios focales permiten comparar defensas dentro de contratos concretos de amenaza, superficie, atacante y baseline. La síntesis se distribuye en Herramientas, permisos y aislamiento: 29/50; Jailbreak y cumplimiento de políticas: 8/50; Contexto, RAG y prompt injection indirecta: 8/50; Exfiltración, secretos y privacidad: 4/50; Monitorización, verificación y salida: 1/50; esa distribución muestra que filtrar prompts, proteger RAG, restringir herramientas, aislar ejecución y verificar salidas resuelven problemas distintos (Huang et al., 2026; L. Zhao et al., 2026; Yung et al., 2025; Z. Lin et al., 2026).

El primer patrón es la incompletitud del contrato comparativo. La amenaza es recuperable en 50/50 estudios, el control en 50/50, el punto de aplicación en 50/50 y un baseline en 45/50. Cuando una evaluación omite cualquiera de estas piezas, la tasa de éxito o bloqueo no permite saber si la defensa mejora una alternativa razonable bajo el mismo riesgo.

El segundo patrón separa eficacia local y robustez. 30/50 estudios reportan ASR o una métrica defensiva equivalente, pero solo 15/50 incluyen señal de atacante adaptativo y 49/50 aportan transferencia, ataques no vistos, ablación o validación externa. La evidencia favorece cautela: vencer un conjunto fijo de prompts demuestra cobertura del conjunto, no resistencia frente a un adversario que conoce y optimiza contra el control.

El tercer patrón es el coste oculto de la seguridad. Falsos positivos aparecen en 13/50 estudios, utilidad en 31/50, latencia en 9/50 y coste en 4/50. Esta asimetría impide aceptar un ranking basado solo en bloqueo: una defensa demasiado restrictiva puede reducir ataques y, al mismo tiempo, impedir uso legítimo, aumentar revisión humana o hacer inviable el sistema.

El cuarto patrón es de cobertura de superficie. Permisos y sandbox actúan sobre herramientas; separación de instrucciones y datos sobre contexto; monitores de runtime sobre trayectorias; y verificadores de salida sobre respuestas. La evidencia no demuestra que más capas sean siempre mejores ni que una barrera única sea necesariamente inferior. Sí demuestra que estos controles no son intercambiables: una comparación válida debe indicar qué ruta de ataque interrumpe cada mecanismo y qué daño residual queda fuera de su alcance.

El quinto patrón es la transparencia del fallo. 41/50 estudios declaran modos de fallo y 38/50 código o artefactos. Estos datos son científicamente centrales: permiten definir la frontera de uso seguro, reproducir bypasses y decidir qué capa compensatoria hace falta. Una defensa que solo publica éxitos aporta menos a la seguridad acumulativa que otra que muestra dónde deja de funcionar.

Tabla 8C. Contrato comparativo para decidir si un harness aporta una mejora defensiva.

Dimensión	Cobertura focal	Regla de interpretación
Amenaza y superficie	50/50	La eficacia solo viaja a amenazas y superficies equivalentes.
Control y enforcement	Control: 50/50; enforcement: 50/50	Debe saberse qué decisión controla y antes de qué daño.
Baseline y eficacia	Baseline: 45/50; eficacia: 30/50	Sin comparación común no existe mejora atribuible.

Dimensión	Cobertura focal	Regla de interpretación
Adaptación y robustez	Atacante adaptativo: 15/50; robustez amplia: 49/50	Separa prueba adversarial adaptativa de transferencia, ablación o ataques no vistos.
Utilidad y coste	Utilidad: 31/50; latencia: 9/50; coste: 4/50	Evita llamar mejor a una defensa inutilizable o demasiado cara.
Fallo y reproducibilidad	Fallo: 41/50; artefacto: 38/50	Delimita riesgo residual y permite repetir la evaluación.

La síntesis permite una decisión fuerte pero condicionada: no hay un campeón universal; hay configuraciones que dominan dentro de una amenaza y un coste de fallo definidos. Cuando dos harnesses intercambian seguridad por utilidad o coste, la literatura debe reportar una frontera de compromiso y no ocultarla detrás de una media única.

Este bloque evita que la síntesis dependa de una frase por familia. El cuerpo principal conserva las señales necesarias para interpretar constructos, mecanismos, unidades de análisis, métodos, comparadores y dirección de resultados; la matriz suplementaria mantiene el detalle por estudio para trazabilidad y actualización.

La matriz completa por estudio no se fuerza dentro del cuerpo del PDF porque pierde legibilidad cuando combina DOI, título, constructos, unidades de análisis, instrumentos, métodos y hallazgos de 50 estudios focales. Por ese motivo, el cuerpo principal resume la información comparable y el material suplementario conserva la trazabilidad detallada.

Tabla 9. Evaluación crítica de reporting, trazabilidad y cautela inferencial.

Diseño	N	Score crítico medio	Perfil de reporting
Experimental	47	81,7	Sin tamaño/corpus: 25; sin contexto: 19; sin teoría: 30; riesgo: Medio
Cuantitativo	2	75,0	Sin tamaño/corpus: 2; sin contexto: 1; sin teoría: 1; riesgo: Medio-bajo

La Tabla 9 resume el perfil agregado, pero la evaluación crítica no queda solo en un promedio ni replica mecánicamente la confianza de extracción. Para cada uno de los 50 estudios focales se aplicó una rúbrica específica de esta revisión para valorar reporting, trazabilidad y cautela inferencial, basada en cinco señales auditables: tamaño muestral o corpus, contexto o país, marco teórico, comparador o línea base y validación/robustez recuperable desde el PDF. El score crítico se calcula como $0,55 \times \text{confianza de extracción} + 0,45 \times \text{cobertura de señales aplicables}$. La cobertura de señales aplicables se obtiene con los pesos de la Tabla 9A; cuando una dimensión no aplica por tipo de trabajo, no penaliza el denominador. Esta rúbrica no se presenta como sustituto universal de una herramienta validada de riesgo de sesgo; separa la calidad de reporte/extractabilidad de la certeza causal. Por eso los resultados empíricos se interpretan con cautela cuando faltan comparador, medición, contexto, validación externa o control de endogeneidad. Los conteos agregados y la codificación de cada estudio pueden recomputarse en [paper/appendices/data/critical-appraisal-matrix.csv](#), donde se conservan DOI, señales reportadas, vacíos, dimensiones no aplicables y score.

La evaluación crítica se interpreta con una rúbrica específica de seguridad. Para cada estudio se comprueba si la amenaza está definida, si el atacante es estático o adaptativo, si existe baseline, si el corpus de ataques es reproducible, si se reporta eficacia defensiva, si se conserva utilidad, si se cuantifican falsos positivos, latencia o coste, y si se prueban ataques no vistos, transferencia entre modelos o modos de fallo. Esta adaptación impide que una reducción local de ASR se trate como evidencia suficiente de superioridad operacional.

Tabla 9A. Pesos de la evaluación crítica de reporting y trazabilidad.

Señal auditable	Peso	Criterio de lectura
Tamaño muestral o corpus	20	Participantes, usuarios, mensajes, países, artículos, casos u otra unidad analítica explícita; no aplica a trabajos no empíricos.
Contexto o país	15	Localización, plataforma, institución, país, región o entorno empírico suficiente para valorar transferencia.
Marco teórico	20	Teoría, modelo conceptual, constructos o literatura de referencia que justifica la relación analizada.
Comparador o línea base	20	Grupo, periodo, plataforma, condición, benchmark, contraste o regla equivalente de comparación.
Validación o robustez	25	Pruebas de robustez, replicación, validez externa, sensibilidad, triangulación o estrategia causal; no aplica a trabajos no empíricos.

Tabla 9B. Riesgo de reporting por estudio focal.

#	Referencia	DOI	Riesgo	Base del juicio
1	Zhu (2025)	10.48550/arxiv.2502.05174	Medio-bajo	Diseño: Experimental. Score crítico 83/100; confianza de extracción 85; vacíos: teoría.
2	Sharma (2025)	10.48550/arxiv.2501.18837	Medio-bajo	Diseño: Experimental. Score crítico 82/100; confianza de extracción 95; vacíos: contexto, teoría.
3	Louck (2026)	10.48550/arxiv.2606.24322	Bajo	Diseño: Experimental. Score crítico 90/100; confianza de extracción 95; vacíos: contexto.
4	Liang (2025)	10.48550/arxiv.2512.06716	Medio-bajo	Diseño: Experimental. Score crítico 84/100; confianza de extracción 88; vacíos: tamaño/corpus.

#	Referencia	DOI	Riesgo	Base del juicio
5	Yung (2025)	10.48550/arxiv.2503.03502	Medio-bajo	Diseño: Experimental. Score crítico 80/100; confianza de extracción 80; vacíos: tamaño/corpus.
6	Ji (2025)	10.48550/arxiv.2511.15203	Medio-bajo	Diseño: Revisión / no aplica diseño empírico. Score crítico 81/100; confianza de extracción 95; vacíos: teoría.
7	Zhang (2026)	10.48550/arxiv.2602.22724	Bajo	Diseño: Cuantitativo. Score crítico 86/100; confianza de extracción 90; vacíos: tamaño/corpus.
8	Zhao (2026)	10.48550/arxiv.2605.17986	Medio-bajo	Diseño: Experimental. Score crítico 82/100; confianza de extracción 95; vacíos: contexto, teoría.
9	Alizadeh (2025)	10.48550/arxiv.2506.01055	Bajo	Diseño: Experimental. Score crítico 88/100; confianza de extracción 95; vacíos: teoría.
10	Dingeto (2026)	10.48550/arxiv.2606.02240	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 90; vacíos: contexto, teoría.
11	Fan (2026)	10.48550/arxiv.2605.11039	Bajo	Diseño: Experimental. Score crítico 88/100; confianza de extracción 95; vacíos: tamaño/corpus.

#	Referencia	DOI	Riesgo	Base del juicio
12	Das (2026)	10.48550/arxiv. 2605.01970	Medio-bajo	Diseño: Experimental. Score crítico 76/100; confianza de extracción 90; vacíos: tamaño/corpus, teoría.
13	Ying (2026)	10.48550/arxiv. 2604.24118	Bajo	Diseño: Experimental. Score crítico 88/100; confianza de extracción 95; vacíos: tamaño/corpus.
14	Wang (2026)	10.48550/arxiv. 2605.26497	Medio	Diseño: Experimental. Score crítico 72/100; confianza de extracción 95; vacíos: tamaño/corpus, contexto, teoría.
15	Huang (2026)	10.48550/arxiv. 2603.21642	Medio	Diseño: Experimental. Score crítico 65/100; confianza de extracción 90; vacíos: teoría, comparador, validación.
16	He (2026)	10.48550/arxiv. 2603.10749	Bajo	Diseño: Experimental. Score crítico 88/100; confianza de extracción 95; vacíos: tamaño/corpus.
17	Li (2026)	10.48550/arxiv. 2601.10173	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 95; vacíos: tamaño/corpus, teoría.

#	Referencia	DOI	Riesgo	Base del juicio
18	Xing (2025)	10.48550/arxiv.2508.10991	Bajo	Diseño: Experimental. Score crítico 97/100; confianza de extracción 95; vacíos: sin vacíos críticos.
19	Betser (2026)	10.48550/arxiv.2601.12449	Medio-bajo	Diseño: Experimental. Score crítico 76/100; confianza de extracción 85; vacíos: tamaño/corpus, contexto.
20	Cunningham (2026)	10.48550/arxiv.2601.04603	Medio	Diseño: Experimental. Score crítico 70/100; confianza de extracción 90; vacíos: tamaño/corpus, contexto, teoría.
21	Lin (2026)	10.48550/arxiv.2605.10582	Bajo	Diseño: Experimental. Score crítico 86/100; confianza de extracción 90; vacíos: tamaño/corpus.
22	Gowda (2026)	10.48550/arxiv.2605.03482	Bajo	Diseño: Experimental. Score crítico 97/100; confianza de extracción 95; vacíos: sin vacíos críticos.
23	Pai (2026)	10.48550/arxiv.2606.17467	Medio-bajo	Diseño: Experimental. Score crítico 82/100; confianza de extracción 95; vacíos: contexto, teoría.
24	Derya (2026)	10.48550/arxiv.2605.03095	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 95; vacíos: tamaño/corpus, teoría.

#	Referencia	DOI	Riesgo	Base del juicio
25	SingGuard Team (2026)	10.48550/arxiv.2607.13081	Bajo	Diseño: Experimental. Score crítico 90/100; confianza de extracción 95; vacíos: contexto.
26	Shi (2026)	10.48550/arxiv.2606.20922	Medio	Diseño: Experimental. Score crítico 72/100; confianza de extracción 95; vacíos: tamaño/corpus, contexto, teoría.
27	Saravi (2026)	10.21203/rs.3.rs-9932271/v1	Bajo	Diseño: Experimental. Score crítico 88/100; confianza de extracción 95; vacíos: teoría.
28	Gupta (2025)	10.48550/arxiv.2508.09288	Medio	Diseño: Experimental. Score crítico 72/100; confianza de extracción 95; vacíos: tamaño/corpus, contexto, teoría.
29	Wang (2025)	10.48550/arxiv.2510.19169	Medio	Diseño: Cuantitativo. Score crítico 64/100; confianza de extracción 80; vacíos: tamaño/corpus, contexto, teoría.
30	Zhao (2026)	10.48550/arxiv.2604.11790	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 95; vacíos: tamaño/corpus, teoría.
31	Chen (2026)	10.48550/arxiv.2604.07223	Bajo	Diseño: Experimental. Score crítico 88/100; confianza de extracción 95; vacíos: teoría.

#	Referencia	DOI	Riesgo	Base del juicio
32	Shi (2025)	10.48550/arxiv.2507.15219	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 95; vacíos: tamaño/corpus, teoría.
33	Jamshidi (2025)	10.48550/arxiv.2512.06556	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 90; vacíos: contexto, teoría.
34	Isbarov (2026)	10.48550/arxiv.2602.05066	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 90; vacíos: contexto, teoría.
35	He (2026)	10.48550/arxiv.2606.15441	Medio-bajo	Diseño: Experimental. Score crítico 76/100; confianza de extracción 90; vacíos: tamaño/corpus, teoría.
36	Yu (2026)	10.48550/arxiv.2601.04795	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 95; vacíos: tamaño/corpus, teoría.
37	Deep (2026)	10.48550/arxiv.2604.23887	Medio-bajo	Diseño: Experimental. Score crítico 82/100; confianza de extracción 95; vacíos: contexto, teoría.

#	Referencia	DOI	Riesgo	Base del juicio
38	Wang (2026)	10.48550/arxiv.2602.20708	Medio	Diseño: Experimental. Score crítico 64/100; confianza de extracción 80; vacíos: tamaño/corpus, contexto, teoría.
39	Thornton (2026)	10.48550/arxiv.2601.03300	Bajo	Diseño: Experimental. Score crítico 90/100; confianza de extracción 95; vacíos: contexto.
40	Zhu (2026)	10.48550/arxiv.2604.03870	Bajo	Diseño: Experimental. Score crítico 86/100; confianza de extracción 90; vacíos: tamaño/corpus.
41	Goren (2026)	10.1609/aaai.v40i44.41074	Medio-bajo	Diseño: Experimental. Score crítico 76/100; confianza de extracción 90; vacíos: tamaño/corpus, teoría.
42	Sharma (2026)	10.21203/rs.3.rs-10196595/v1	Bajo	Diseño: Experimental. Score crítico 90/100; confianza de extracción 95; vacíos: contexto.
43	Kang (2025)	10.48550/arxiv.2512.00966	Medio	Diseño: Experimental. Score crítico 70/100; confianza de extracción 90; vacíos: tamaño/corpus, contexto, teoría.
44	Bhagwatkar (2025)	10.48550/arxiv.2510.05244	Medio-bajo	Diseño: Experimental. Score crítico 83/100; confianza de extracción 85; vacíos: tamaño/corpus.

#	Referencia	DOI	Riesgo	Base del juicio
45	Zha (2026)	10.48550/arxiv. 2605.02812	Bajo	Diseño: Experimental. Score crítico 94/100; confianza de extracción 90; vacíos: sin vacíos críticos.
46	Zou (2026)	10.48550/arxiv. 2602.10915	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 95; vacíos: tamaño/corpus, teoría.
47	Ge (2026)	10.48550/arxiv. 2603.07191	Bajo	Diseño: Experimental. Score crítico 88/100; confianza de extracción 95; vacíos: teoría.
48	Wang (2026)	10.48550/arxiv. 2603.28807	Bajo	Diseño: Experimental. Score crítico 86/100; confianza de extracción 90; vacíos: tamaño/corpus.
49	Li (2024)	10.48550/arxiv. 2410.22770	Medio-bajo	Diseño: Experimental. Score crítico 83/100; confianza de extracción 85; vacíos: teoría.
50	Ramakrishnan (2025)	10.48550/arxiv. 2511.15759	Medio-bajo	Diseño: Experimental. Score crítico 79/100; confianza de extracción 90; vacíos: contexto, teoría.

Tabla 9C. Indicadores agregados de evaluación crítica.

Señal auditable	Reportada	Vacío	No aplica	Lectura interpretativa
Tamaño muestral o corpus	22	27	1	Define la unidad empírica real y evita comparar estudios sin base observacional equivalente.
Contexto o país	30	20	0	Permite valorar transferencia entre plataformas, países, instituciones o situaciones históricas.
Marco teórico	18	32	0	Sostiene la acumulación conceptual y evita agrupar hallazgos solo por vocabulario parecido.
Comparador o línea base	49	1	0	Hace posible distinguir señal descriptiva, contraste empírico y afirmación causal prudente.
Validación o robustez	48	1	1	Separa rendimiento local de estabilidad, sensibilidad, réplica o validez externa.

La Tabla 9C resume la codificación crítica de forma agregada para no desplazar al cuerpo principal una matriz excesivamente granular. En esta tabla, la columna reportada indica señales recuperables en el texto completo, la columna vacío indica reporte insuficiente y la columna no aplica identifica dimensiones que no corresponden al tipo de trabajo evaluado.

En conjunto, los resultados se apoyan en lectura de texto completo, extracción estructurada y tablas de síntesis comparables. La síntesis focal permite identificar patrones sin borrar la variabilidad metodológica que caracteriza al corpus (Alizadeh et al., 2025; Gowda, 2026; Xing et al., 2025; Zhang et al., 2026). ## Estructura relacional del corpus

El análisis estructural parte de 50 estudios incluidos, de los que 50 forman el subconjunto focal. Su función no es reemplazar la síntesis sustantiva, sino comprobar si autores, temas, referencias y dimensiones analíticas forman concentraciones, puentes o vacíos que una tabla plana no permite observar.

La red de coautoría tiene estado **interpretable**. Los nodos con mayor conectividad directa son Alex Silverstein, Alwin Peng, Andy Dau, Clare O'Hara y Ethan Perez. Esta posición describe colaboración dentro del corpus, no calidad ni autoridad. La red de descriptores tiene estado **exploratorio** y no alcanza estabilidad

suficiente para sostener un núcleo temático consolidado; su interpretación debe mantenerse dentro de la cobertura de palabras clave reportada (42.0%).

El acoplamiento bibliográfico tiene estado **exploratorio** y cobertura de referencias del 0.0%. Cuando esa cobertura o la estabilidad de comunidades no alcanza el umbral, el patrón se conserva como señal exploratoria y no como prueba de escuelas consolidadas. El atlas HTML permite contrastar estas relaciones por fase de selección y consultar las definiciones de cada indicador.

Discusión

La seguridad de modelos generativos y sistemas agénticos debe entenderse como una propiedad del sistema completo y de su entorno de amenaza, no como una propiedad intrínseca del modelo ni como la mera presencia de un filtro (Huang et al., 2026; L. Zhao et al., 2026; Yung et al., 2025; Z. Lin et al., 2026).

El primer punto de discusión es que la evidencia focal se distribuye entre Herramientas, permisos y aislamiento: 29/50; Jailbreak y cumplimiento de políticas: 8/50; Contexto, RAG y prompt injection indirecta: 8/50; Exfiltración, secretos y privacidad: 4/50; Monitorización, verificación y salida: 1/50. Esta composición muestra que **harness de seguridad** designa controles distintos: algunos filtran lenguaje, otros aíslan contexto, restringen herramientas, monitorizan ejecución, verifican salidas o protegen secretos. Compararlos como una sola clase borraría qué superficie protege cada uno y qué daño puede seguir ocurriendo (Huang et al., 2026; L. Zhao et al., 2026; Yung et al., 2025; Z. Lin et al., 2026).

Las familias sostenidas por un único estudio —Monitorización, verificación y salida— se conservan para no borrar una superficie emergente, pero no se usan para inferir madurez, eficacia típica ni prioridad de despliegue. Su función es señalar una hipótesis que requiere réplica, no añadir peso a una mayoría defensiva.

El segundo punto es la comparabilidad. En 50/50 estudios se recupera una amenaza explícita, en 50/50 un punto de aplicación y en 45/50 un baseline. Cuando falta una de estas piezas, una cifra de eficacia pierde parte de su significado: no sabemos frente a qué atacante funciona, qué decisión controla ni si mejora una alternativa razonable. Por eso la revisión no construye un ranking universal; construye comparaciones condicionadas por contrato de amenaza (Alizadeh et al., 2025; Gowda, 2026; Xing et al., 2025; Zhang et al., 2026).

El tercer punto es la adaptatividad. 15/50 estudios reportan señal de atacante adaptativo y 49/50 aportan robustez, transferencia, ablación o ataques no vistos. La segunda cifra es deliberadamente más amplia y no implica que todos esos estudios enfrenten a un adversario adaptativo: también incluye transferencia entre modelos o dominios, ablaciones y conjuntos no vistos. Una defensa evaluada sobre prompts estáticos puede aprender la forma del benchmark sin resistir a un adversario que conoce el filtro, reformula el ataque o desplaza la inyección a documentos, memoria o herramientas. La diferencia entre cobertura estática y robustez adaptativa es la frontera más importante entre demostración y seguridad operacional (Dingeto & Leeney, 2026; Louck, 2026; Pai, 2026; Ying et al., 2026).

El cuarto punto es el coste de proteger. Solo 13/50 estudios reportan falsos positivos, 31/50 utilidad, 9/50 latencia y 4/50 coste. Una defensa puede reducir la tasa de ataque porque rechaza demasiadas entradas, limita herramientas legítimas o añade una revisión costosa. La superioridad científica exige medir simultáneamente seguridad y capacidad útil; de lo contrario, el problema se resuelve desactivando parte del sistema que se pretendía proteger. La baja cobertura de latencia y coste impide convertir las fronteras observadas en una recomendación de producción: sin cómputo adicional, tokens, tiempo de respuesta, intervención humana y coste de falsos rechazos no puede estimarse el coste total de operar la defensa. La consecuencia es metodológica y empresarial: una frontera de dominancia sin esas métricas sigue siendo una hipótesis comparativa útil, pero no una decisión de despliegue cerrada.

El quinto punto es que las capas defensivas no son intercambiables. Un filtro de entrada puede detener patrones conocidos, pero no controla una llamada de herramienta ya autorizada; un sandbox limita impacto, pero no evita fuga de información en la respuesta; un verificador de salida puede detectar daño, pero llega

tarde si el agente ya ejecutó una acción irreversible. La arquitectura adecuada depende del momento en que el daño puede materializarse y del coste de una falsa aceptación frente a un falso rechazo.

La concentración de configuraciones en controles no tipificados no constituye una familia defensiva adicional ni una réplica implícita. Es un diagnóstico de inmadurez taxonómica: parte de la literatura usa etiquetas genéricas, combina mecanismos sin aislar su efecto o no describe con precisión qué transición del sistema interrumpe. Mientras esa opacidad persista, sumar esos estudios como si probaran el mismo control produciría una mayoría artificial y debilitaría, en lugar de fortalecer, la comparación.

Finalmente, la madurez del campo depende de sus fallos visibles. 41/50 estudios reportan modos de fallo y 38/50 código o artefactos recuperables. Los resultados negativos, bypasses y límites de transferencia no debilitan una defensa; permiten delimitar dónde puede usarse. Un harness científicamente útil no promete invulnerabilidad: ofrece una reducción de riesgo verificable, declara qué deja pasar y facilita repetir la prueba.

Implicaciones teóricas

La primera implicación teórica derivada de los harnesses analizados es que su seguridad no puede localizarse únicamente en el modelo base. Dentro del alcance de este corpus, el comportamiento seguro emerge de una relación entre modelo, contexto, herramientas, memoria, permisos, verificadores y entorno de ejecución. Esta perspectiva no se presenta como una verdad ontológica sobre todo sistema de IA: es una inferencia arquitectónica apoyada por las defensas operacionales revisadas. La unidad teórica relevante deja de ser la respuesta aislada del modelo y pasa a ser el circuito que transforma información en una capacidad, una decisión y una consecuencia (Huang et al., 2026; L. Zhao et al., 2026; Yung et al., 2025; Z. Lin et al., 2026).

La segunda implicación es que una defensa se define por el daño que interrumpe y por el momento en que lo hace. Filtrar una entrada, separar instrucciones de datos, negar una capacidad, aislar una ejecución y verificar una salida son mecanismos causalmente distintos. La teoría del campo debe explicar qué ruta de ataque corta cada control, no limitarse a contar cuántas capas existen. Esto introduce una gramática causal mínima: autoridad disponible, capacidad solicitada, transición permitida, evidencia usada para decidir y contención posterior al fallo.

La tercera implicación es que detección y enforcement no son sustitutos. Un clasificador puede reconocer una intención dañina sin impedir que una herramienta actúe; una política de permisos puede contener la acción aunque no identifique semánticamente el ataque. La distinción permite ordenar dos tradiciones que a menudo se comparan de manera impropia: las defensas epistemológicas, que estiman si una entrada o trayectoria es peligrosa, y las defensas de autoridad, que limitan lo que el sistema puede hacer incluso cuando la estimación falla. En sistemas de alto impacto, ambas funciones son complementarias y su composición debe evaluarse como tal (Alizadeh et al., 2025; Gowda, 2026; Xing et al., 2025; Zhang et al., 2026).

La cuarta implicación es que robustez y cobertura no son equivalentes. Cobertura describe rendimiento sobre un conjunto conocido; robustez exige mantener la reducción de riesgo cuando cambia el modelo, el dominio, la formulación o la estrategia del atacante. Esta distinción impide interpretar buenos resultados estáticos como seguridad general. Un atacante adaptativo no es solo una condición experimental más exigente: es una prueba epistemológica de si el mecanismo defensivo captura la ruta de daño o únicamente regularidades superficiales del benchmark.

La quinta implicación es multiobjetivo: seguridad, utilidad, disponibilidad, latencia, coste y operabilidad forman una frontera de decisión. No existe una defensa universalmente mejor cuando una alternativa bloquea más ataques pero degrada de forma material el uso legítimo o desplaza el coste hacia revisión humana. Teóricamente, la dominancia entre harnesses es un orden parcial: una configuración solo puede dominar a otra dentro de un contrato equivalente y cuando no empeora de forma sustantiva las dimensiones operacionales relevantes. Fuera de esas condiciones, una media única destruye información de decisión.

La sexta implicación es que el fallo residual forma parte de la teoría de la defensa. Un harness maduro no se define por prometer bloqueo total, sino por delimitar qué ataques reduce, cuáles siguen siendo posibles, cómo se detecta el fallo y qué mecanismo contiene el daño después de una evasión. Esta lectura desplaza la seguridad desde la fantasía de invulnerabilidad hacia una teoría de reducción, observabilidad y contención del

riesgo. Publicar bypasses, falsos negativos y condiciones de degradación no debilita la contribución; revela su frontera de validez (Dingeto & Leeney, 2026; Louck, 2026; Pai, 2026; Ying et al., 2026).

La séptima implicación afecta a la composición de capas. Añadir controles no garantiza una mejora monótona: dos filtros pueden compartir el mismo punto ciego, una verificación tardía puede no reparar una acción irreversible y una política restrictiva puede trasladar el riesgo a operadores humanos que aprueban excepciones bajo presión. La teoría debe estudiar independencia de fallos, orden de ejecución y capacidad de recuperación, no solo número de componentes. Una defensa multicapa aporta valor cuando cada capa cubre un modo de fallo distinto y la interacción entre ellas conserva una ruta de decisión interpretable.

La octava implicación es acumulativa. El campo solo podrá comparar resultados si comparte una gramática mínima: amenaza, atacante, superficie, control, enforcement, baseline, eficacia, utilidad, coste y robustez. Esa gramática permite integrar estudios heterogéneos sin convertirlos en un ranking engañoso. También convierte los vacíos de reporte en información teórica: si no se declara quién ataca, qué autoridad posee el sistema o qué daño se contiene, la afirmación de seguridad carece de objeto estable.

La novena implicación es una regla de diseño científico. Los estudios futuros deberían formular la defensa como una hipótesis refutable sobre una transición concreta del sistema: bajo una amenaza definida, el control impide o contiene una acción sin superar un coste operacional preestablecido. Esa formulación obliga a declarar qué resultado refutaría la propuesta y permite que una réplica modifique modelo, dominio o atacante sin perder la unidad comparativa. La teoría de los harnesses avanzará cuando pueda explicar no solo que una defensa funcionó, sino por qué funcionó, dónde deja de hacerlo y qué evidencia permitiría sustituirla (Cunningham et al., 2026; Derya & Sunar, 2026; Sharma et al., 2025; Y. He et al., 2026).

Implicaciones prácticas

Para equipos que despliegan modelos y agentes, la implicación práctica es seleccionar controles desde el daño posible y el punto donde puede materializarse, no desde una lista genérica de guardrails.

La primera consecuencia es escribir el threat model antes de elegir el control. Prompt injection directo, inyección indirecta desde RAG, jailbreak, abuso de herramientas y exfiltración no son variantes intercambiables del mismo problema. Un filtro de entrada puede ser pertinente para una superficie y completamente ciego para otra; por eso una política de seguridad debe declarar qué activo protege, qué puede hacer el atacante y dónde se aplica la defensa.

La segunda consecuencia es exigir un baseline material. En 45/50 estudios focales se recupera un comparador explícito. Sin ejecución sin defensa, control alternativo o configuración ablation, una tasa de ataque no permite saber cuánto valor añade el harness ni qué componente produce el efecto.

La tercera consecuencia es probar atacantes adaptativos. Solo 15/50 estudios dejan una señal recuperable de adaptación del atacante. Una defensa evaluada únicamente contra un conjunto estático puede estar midiendo memorización del benchmark, no resistencia. Antes de desplegar, el equipo debería retestar después de revelar reglas, mensajes de rechazo, herramientas y superficie de contexto.

La cuarta consecuencia es medir seguridad y utilidad en el mismo experimento. 30/50 estudios reportan ASR o una métrica equivalente, pero solo 31/50 dejan impacto de utilidad y 13/50 falsos positivos recuperables. Bloquear más no es dominar si el sistema también rechaza tareas legítimas, degrada precisión o desplaza la carga a revisión humana.

La quinta consecuencia es presupuestar el control como parte de la arquitectura. La latencia aparece en 9/50 estudios y el coste en 4/50. En producción deben medirse por transacción, herramienta y nivel de riesgo; de lo contrario una defensa técnicamente eficaz puede ser operativamente inviable o inducir atajos que la neutralicen.

La sexta consecuencia es diseñar por capas solo cuando cada capa tiene una función verificable. Entrada, contexto, runtime, permisos de herramientas y salida cubren superficies distintas. La defensa en profundidad aporta valor cuando reduce rutas de evasión y conserva observabilidad; añadir detectores redundantes sin ablation solo aumenta complejidad y falsos positivos.

La séptima consecuencia es convertir fallos y robustez en requisitos de compra. 49/50 estudios aportan alguna prueba de robustez y 41/50 explicitan modos de fallo. Un proveedor o equipo interno debería entregar límites conocidos, ataques que siguen funcionando, política de actualización, logs auditables y procedimiento de rollback, no solo una media de benchmark.

La octava consecuencia es exigir reproducibilidad proporcional al riesgo. 38/50 estudios declaran código, datos o artefactos recuperables. Cuando el control decide si un agente puede leer datos, llamar herramientas o ejecutar acciones, el contrato de adopción debería incluir configuración, versiones, conjuntos de prueba, umbrales y evidencia de regresión.

En términos sustantivos, la revisión permite identificar configuraciones defensivas prometedoras, pero solo puede llamar mejor a una alternativa cuando comparte amenaza y baseline y conserva utilidad con coste y fallo residual explícitos.

Aportación original del artículo

La aportación central del artículo es rechazar la idea de que un harness es mejor porque bloquea más ataques en un benchmark aislado. Una defensa solo puede considerarse superior dentro de un contrato operacional explícito: amenaza y atacante definidos, superficie protegida, punto de aplicación, baseline comparable, reducción de riesgo, utilidad preservada, coste asumible y fallo residual conocido.

La tesis teórica es que la seguridad de modelos generativos y agentes no reside en una barrera única, sino en la distribución de control a lo largo del sistema. Filtros de entrada, aislamiento de contexto, permisos de herramientas, monitores de ejecución, verificadores de salida y auditoría reducen incertidumbres distintas. Sumar capas no garantiza seguridad: cada capa cambia la superficie de ataque, introduce falsos positivos y puede crear puntos ciegos nuevos.

El modelo interpretativo propuesto es amenaza-superficie-control-evidencia-coste. La amenaza declara qué capacidad tiene el atacante; la superficie identifica dónde puede influir; el control especifica qué decisión se bloquea, limita o verifica; la evidencia mide el riesgo residual frente a baselines y ataques adaptativos; y el coste incorpora utilidad, latencia, cómputo y carga de operación. La comparación pierde validez cuando omite cualquiera de esas piezas.

Tabla 10. Modelo de aportación original del artículo.

Plano	Tesis del artículo	Valor para el campo
Unidad de teoría	Contrato operacional de defensa	Impide comparar controles bajo amenazas o costes incompatibles.
Mecanismo central	Amenaza-superficie-control-evidencia-coste	Explica dónde se reduce riesgo y dónde se desplaza.
Aporte disciplinar	Frontera de dominancia defensiva	Sustituye un ranking único por comparaciones multiobjetivo auditables.
Regla futura	Evaluar adaptación, utilidad y fallo residual	Evita declarar victoria sobre benchmarks estáticos o defensas demasiado restrictivas.

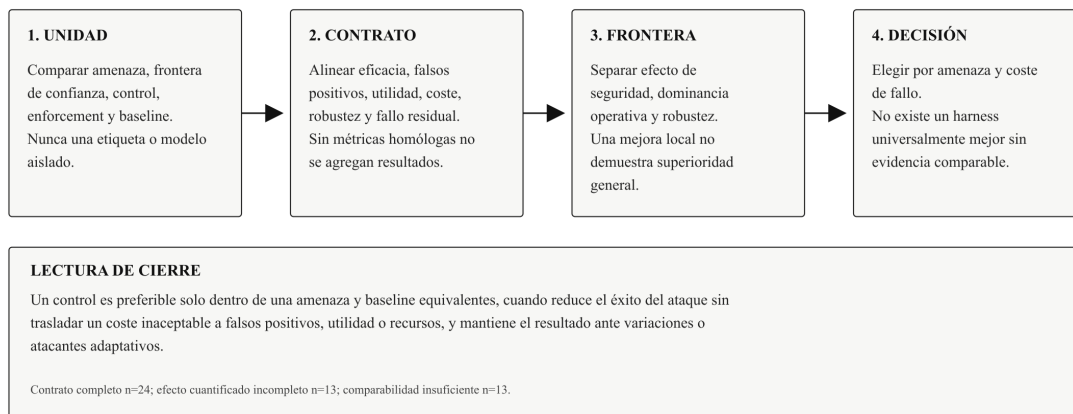
El aporte al campo es una regla de decisión que sustituye rankings universales por fronteras de dominancia. Un harness domina a otro solo si reduce más riesgo bajo la misma amenaza y baseline sin empeorar de manera material utilidad, falsos positivos, coste o latencia; cuando existen compensaciones, la conclusión correcta es contextual y debe nombrar qué propiedad se prioriza.

La consecuencia metodológica es exigir que futuras evaluaciones reporten threat model, atacante adaptativo o estático, punto de enforcement, corpus de ataques, baseline, ASR o métrica equivalente, falsos positivos,

utilidad, latencia, coste, robustez fuera de distribución y modos de fallo. Sin esa ficha mínima, el estudio puede demostrar una técnica, pero no sostener que ofrece el mejor harness.

Gramática analítica de la revisión

Modelo interpretativo para convertir la revisión sobre Harnesses de seguridad para modelos generativos y sistemas agénticos en una gramática comparativa.



Base: 50 estudios focales; figura de aportación derivada de extracción estructurada y síntesis interpretativa.

Figura 6: Modelo interpretativo de la gramática analítica propuesta por la revisión.

La Figura 6 sintetiza la aportación del artículo como una gramática comparativa: define la unidad real de comparación, ordena las dimensiones que explican el campo, separa evidencia estable de señal emergente y convierte los vacíos de reporte en agenda futura.

Amenazas a la validez

Esta sección no se plantea como una disculpa metodológica, sino como delimitación explícita del alcance de la síntesis. Una revisión sistemática gana fuerza cuando dice con precisión qué puede sostener, qué solo aparece como señal y qué queda fuera de sus condiciones de evidencia.

La primera amenaza afecta a la selección y cobertura. A partir de esos 116 estudios incluidos, se definió una síntesis focal de 50 estudios para la comparación intensiva. Los 66 estudios restantes siguieron dentro del corpus de revisión como perímetro contextual elegible: delimitan el campo y conservan trazabilidad, pero no alcanzaron el umbral combinado de ajuste, calidad, representatividad y densidad extractiva que exige la comparación intensiva. Por eso se reportan como contexto auditable y no como evidencia fina del N focal. El subconjunto focal de n=50 no es una muestra aleatoria del campo, sino una selección intensiva basada en ajuste temático, densidad de extracción, PDF legible y score compuesto. Esta decisión aumenta la calidad analítica, pero puede concentrar estudios mejor reportados o más fáciles de extraer.

La segunda amenaza procede de la recuperabilidad del texto completo. De 200 textos completos buscados, 57 no pudieron recuperarse como PDF legible y quedaron fuera por regla metodológica. Esa decisión mejora auditabilidad y evita inferencias desde registros incompletos, pero puede sesgar el corpus hacia publicaciones con DOI, acceso abierto, mejor indexación o PDF técnicamente recuperable.

La tercera amenaza es inferencial. La amenaza inferencial principal es convertir patrones descriptivos en conclusiones causales. La revisión identifica señales, condiciones y vacíos, pero no debe leerse como meta-análisis causal cuando los diseños originales no son commensurables (Dingeto & Leeney, 2026; Louck, 2026; Pai, 2026; Ying et al., 2026).

La cuarta amenaza afecta a la validez de constructo. La validez de constructo exige cautela porque Harnesses de seguridad para modelos generativos y sistemas agénticos puede aparecer operacionalizado mediante diseños, unidades de análisis, instrumentos y resultados no equivalentes. La revisión mitiga ese riesgo usando una gramática común de comparación, pero no borra la heterogeneidad del material fuente.

La quinta amenaza es de reporting y extracción. La revisión depende de lo que los artículos hacen explícito: 32 estudios no declaran marco teórico suficiente, 18 estudios no detallan muestra, 20 estudios no detallan país o contexto y 5 estudios no explicitan variables o dimensiones analíticas. Estos vacíos no invalidan automáticamente el corpus, pero sí limitan la fuerza de las inferencias.

La sexta amenaza es la equivalencia entre unidades de análisis. Aunque el subconjunto focal no muestra una concentración clara de estudios de unidad mínima o casos aislados, la comparación sigue dependiendo de que cada trabajo declare población, nivel de análisis, contexto y alcance inferencial de forma suficiente.

La séptima amenaza es temporal. En campos rápidos, una ventana de publicación puede capturar un frente técnico todavía inestable: preprints, prototipos, benchmarks tempranos, versiones de modelos que cambian y prácticas de reporte no consolidadas. Por eso la revisión debe leerse como una síntesis fechada de evidencia verificable y no como mapa definitivo de todo el campo.

Limitaciones explícitas del estudio

La primera limitación es que el cribado se apoya en dos juicios automáticos independientes y no en dos revisores humanos ciegos. El acuerdo bruto y el kappa reportados miden consistencia entre esos juicios, no verdad científica. En título y resumen, una discrepancia se conservó como **necesita más prueba** para evitar una exclusión automática; en texto completo, todas las discrepancias recibieron decisión investigadora firmada tras lectura del PDF. La extracción y la evaluación crítica permanecen como codificación asistida y auditable, no como doble codificación humana.

La mitigación aplicada combina protocolo previo, reglas de elegibilidad, juicios separados, resolución conservadora de incertidumbre, adjudicación humana de discrepancias en texto completo, DOI público, PDF local, extracción estructurada, anexos suplementarios, matriz de auditoría, score reproducible y conservación de estudios contextuales fuera del N focal. Estas medidas permiten que un lector externo recomponga cada decisión y detecte dónde podría discrepar, pero no convierten el proceso en un panel humano doble si una revista exige ese diseño específico.

Si la revista objetivo exige doble revisión humana, la validación adicional debería concentrarse en una submuestra aleatoria y heterogénea que cubra extracción de variables, codificación de mecanismos, evaluación crítica y decisión focal; el cribado ya conserva ambos juicios y las decisiones humanas de discrepancia para facilitar esa réplica. El manuscrito no presenta la trazabilidad documental como equivalente a fiabilidad interjueces humana: declara su alcance y deja preparada la evidencia para una verificación posterior.

La segunda limitación es que la rúbrica de score y riesgo de reporting es interna. Se operacionaliza y se publica para que pueda ser auditada, pero no debe presentarse como una herramienta validada universal. Su función es ordenar un corpus heterogéneo y hacer visibles vacíos de reporte; no producir una escala psicométrica generalizable.

Una derivada concreta de esa limitación aparece en el perímetro contextual elegible no focal. Cuando los estudios quedan fuera de la síntesis intensiva, deben conservarse como registros contextuales elegibles y no como estudios con evaluación individual completa. Esa decisión evita sobrerrepresentar una precisión que la extracción no sostiene: el perímetro contextual sirve para trazabilidad, cobertura y actualización, no para afirmar que todos esos estudios tengan exactamente la misma calidad científica.

La tercera limitación es de cobertura. Las fuentes bibliográficas consultadas, los términos de búsqueda, la disponibilidad de APIs, los límites de acceso abierto y la recuperabilidad de PDFs condicionan qué estudios llegan a texto completo. El corpus final es robusto para la pregunta y la ventana temporal, pero no equivale a toda la literatura posible.

La cuarta limitación es de generalización. Los resultados deben transferirse con cautela a otros países, disciplinas, instituciones, niveles educativos, modelos, políticas de datos o prácticas docentes. Una revisión sistemática puede identificar patrones y condiciones de comparación; no puede garantizar que una intervención funcione igual fuera de los contextos reportados por los estudios primarios.

Las mitigaciones aplicadas son: regla DOI/PDF, lectura de texto completo, separación entre corpus incluido y síntesis focal, matriz de selección, extracción estructurada, anexos CSV, sensibilidad del ranking cuando procede, citas trazables y cautela explícita en resultados, discusión y conclusiones. El límite no invalida la síntesis; define las condiciones bajo las cuales puede ser auditada, discutida y actualizada.

Conclusiones

En respuesta a la pregunta de investigación, ¿Qué arquitecturas, controles y estrategias de evaluación de los harnesses de seguridad para modelos generativos y sistemas agénticos reducen de forma más consistente los ataques de prompt injection y jailbreak, el uso inseguro de herramientas, la fuga de datos y las violaciones de políticas, y bajo qué amenazas, diseños comparativos y costes superan a las alternativas o baselines?, la evidencia indica que no existe evidencia suficiente para declarar un harness universalmente mejor. La evidencia permite identificar defensas superiores dentro de comparaciones concretas, pero la superioridad depende de compartir amenaza, atacante, superficie, baseline y criterio de coste, y de demostrar que la reducción del riesgo no se obtiene degradando de forma material la utilidad (Alizadeh et al., 2025; L. Zhao et al., 2026; Yung et al., 2025; Zhang et al., 2026).

El corpus final incluido contiene 116 estudios y la síntesis focal compara 50 configuraciones defensivas con texto completo. 49 estudios son empíricos según la extracción. La conclusión se apoya en esa base comparativa y usa los trabajos teóricos o metodológicos para interpretar mecanismos, no para inflar la evidencia de eficacia.

Esta revisión muestra que un harness de seguridad no debe compararse por nombre, número de filtros o una tasa de ataque aislada, sino por la configuración entre amenaza, superficie protegida, punto de aplicación, adaptatividad del atacante, baseline, reducción del riesgo, pérdida de utilidad y coste operacional.

De esta revisión emerge una gramática de comparación defensiva basada en amenaza, atacante, control, punto de aplicación, baseline, eficacia, falsos positivos, utilidad, latencia, coste, robustez y modo de fallo de los 50 estudios focales.

La distribución observada desplaza la decisión desde el nombre de una defensa hacia la superficie que protege y el daño que contiene. La cobertura desigual de baseline, atacante adaptativo y coste permite priorizar arquitecturas para una amenaza concreta, pero impide convertir resultados locales en una liga única de harnesses (Dingeto & Leeney, 2026; Liang et al., 2025; Louck, 2026; Sharma et al., 2025).

Lo que puede afirmarse con más seguridad es que el harness debe cubrir las fronteras donde datos no confiables pueden convertirse en instrucciones, permisos, llamadas de herramienta o salidas con efecto. Ninguna familia aislada cubre por definición entrada, contexto, memoria, herramientas, runtime y salida; esta es una conclusión arquitectónica, no una afirmación de superioridad empírica entre controles. Lo que aparece como señal emergente es una transición desde guardrails monolíticos hacia defensa en profundidad, pero el corpus todavía no demuestra qué combinación es mejor bajo comparaciones homogéneas. Lo que todavía no puede concluirse es qué combinación domina de forma general: solo 13/50 estudios reportan falsos positivos, 31/50 utilidad, 9/50 latencia y 4/50 coste.

El límite no invalida la síntesis; define su alcance. La regla de DOI público, PDF legible, lectura de texto completo y matriz de selección reduce amplitud potencial, pero aumenta auditabilidad y evita sostener conclusiones sobre registros que no pueden comprobarse desde el documento fuente.

Como aportación final, el artículo propone una frontera de dominancia defensiva para evaluar harnesses de seguridad. Una alternativa domina solo cuando, bajo la misma amenaza y baseline, reduce más riesgo sin empeorar materialmente utilidad, falsos positivos, latencia o coste; si las métricas se compensan, la decisión debe declarar el contexto y la prioridad de riesgo. Esta frontera es una regla teórica derivada del diagnóstico

del corpus, no una jerarquía empírica ya validada entre todos los harnesses. Su función es especificar qué evidencia futura permitiría demostrar dominancia y qué comparaciones deben seguir siendo contextuales. La regla permite transformar resultados fragmentarios de jailbreak, prompt injection, herramientas y exfiltración en decisiones comparables sin fingir una clasificación universal (Das et al., 2026; Fan et al., 2026; Ji et al., 2025; Zhu et al., 2025).

Líneas futuras

Las líneas futuras no se plantean aquí como un cierre ornamental ni como una lista genérica de deseos. En una revisión sistemática, la agenda posterior debe derivarse de los límites observados en la evidencia primaria: qué no se puede comparar todavía, qué se reporta con insuficiente detalle y qué condiciones faltan para convertir señales recurrentes en conocimiento acumulativo. Por eso esta sección no propone simplemente ampliar el número de estudios, sino mejorar la calidad inferencial de los próximos trabajos.

En el subconjunto focal de 50 estudios, los vacíos de reporte permiten ordenar la agenda futura en varios planos complementarios: base teórica, contexto empírico, operacionalización de variables o constructos, equivalencia de comparadores y validación. La lógica es acumulativa: cada línea futura responde a una limitación concreta detectada durante la extracción y señala qué tendría que cambiar en los estudios primarios para que una revisión posterior pueda comparar con más precisión, menor ambigüedad y mayor fuerza explicativa.

En conjunto, estos vacíos señalan un problema de conmensurabilidad: el campo puede producir resultados localmente útiles sin ofrecer todavía las piezas necesarias para acumularlos bajo una explicación compartida.

Una línea futura prioritaria es fortalecer la base teórica de los estudios primarios. En esta revisión, 32 de 50 trabajos focales no declaran una base teórica suficiente para comparación acumulativa; por tanto, la agenda no consiste solo en producir más estudios, sino en explicar con qué marco conceptual se justifican sus unidades de análisis, mecanismos y límites.

Otra línea futura es mejorar la trazabilidad del contexto empírico. 18 estudios no detallan muestra; 20 estudios no detallan país, entorno o contexto territorial; esa ausencia no es un detalle menor, porque limita la posibilidad de saber dónde vale una conclusión, para qué población, con qué unidad de análisis y bajo qué condiciones de transferencia.

Otra línea futura es operacionalizar mejor variables, constructos o dimensiones analíticas. 5 estudios focales no ofrecen un esquema suficientemente explícito; por eso futuras investigaciones deberían declarar qué se mide, qué se manipula o compara, qué resultado cuenta como evidencia y qué dimensión queda solo como interpretación secundaria.

Otra línea futura es evaluar defensas bajo atacantes adaptativos que conozcan el control y puedan optimizar contra él. Una defensa que funciona solo frente a un conjunto estático de prompts demuestra cobertura de benchmark, no robustez operacional.

También se necesitan comparaciones que reporten conjuntamente seguridad y utilidad. Reducir la tasa de ataque bloqueando solicitudes legítimas, degradando la tarea o introduciendo una latencia prohibitiva no demuestra superioridad del sistema; desplaza el riesgo hacia disponibilidad, experiencia de uso o coste.

Aporte teórico e interpretativo del autor

Tesis autoral

El cierre autoral es que una revisión sobre Harnesses de seguridad para modelos generativos y sistemas agénticos solo aporta plenamente cuando transforma evidencia dispersa en una regla de comparación. La fuerza del artículo no está en declarar un consenso rápido, sino en mostrar qué debe estar alineado para que distintos estudios puedan leerse como parte de la misma conversación científica.

Modelo interpretativo propuesto

El modelo de defensa como contrato operacional cumple esa función. Ordena objeto, mecanismo, método, evidencia y límite para distinguir hallazgos acumulables, señales emergentes y afirmaciones que todavía no pueden sostenerse. Así la revisión deja de ser un inventario y se convierte en una infraestructura intelectual para el campo.

La contribución final es metodológica y sustantiva al mismo tiempo: propone cómo pensar el objeto revisado, cómo comparar estudios futuros y cómo convertir los límites de reporte en agenda científica. Una revisión posterior podrá ampliar el corpus, pero no debería volver a empezar sin declarar su unidad de comparación.

Aporte al campo

El modelo se deriva del contraste entre 50 estudios focales, sus hallazgos recuperables y sus límites de reporte; no funciona como tesis externa al corpus, sino como formalización del problema de comparación que la revisión detecta.

Para el campo, el modelo de defensa como contrato operacional funciona como regla de acumulación: obliga a distinguir evidencias que pueden sostener una afirmación común, señales que solo orientan investigación futura y vacíos que impiden cerrar una conclusión fuerte. Esa distinción evita que la revisión se convierta en una suma de resultados compatibles solo en apariencia.

El resultado práctico es un estándar de lectura para trabajos futuros. Un nuevo estudio no debería incorporarse al mismo plano de comparación solo porque use vocabulario parecido; debería mostrar qué unidad teórica trabaja, qué mecanismo defiende, qué evidencia produce, qué límite reconoce y qué parte del modelo confirma, matiza o contradice.

El aporte del autor no se añade como comentario final, sino como regla de interpretación derivada del corpus: qué puede compararse, qué debe mantenerse como señal y qué exige nueva evidencia antes de convertirse en conclusión.

Tabla de contribución teórica e inferencial. Tesis, modelo y regla de acumulación propuestos por el artículo.

Plano	Tesis del autor	Consecuencia científica
Unidad de teoría	Contrato operacional de defensa	Impide comparar controles bajo amenazas o costes incompatibles.
Mecanismo central	Amenaza-superficie-control-evidencia-coste	Explica dónde se reduce riesgo y dónde se desplaza.
Aporte disciplinar	Frontera de dominancia defensiva	Sustituye un ranking único por comparaciones multiobjetivo auditables.
Regla futura	Evaluar adaptación, utilidad y fallo residual	Evita declarar victoria sobre benchmarks estáticos o defensas demasiado restrictivas.
Frontera inferencial	Los vacíos de reporte no se tratan como defectos administrativos, sino como límites de aquello que el campo puede afirmar con rigor.	La cautela deja de ser disculpa y se convierte en criterio de validez.
Producto acumulativo	La matriz, los criterios y los anexos permiten reabrir la revisión sin reiniciar la discusión desde cero.	La revisión queda como infraestructura de contraste, no como cierre retórico.

La base empírica de esta tesis procede de 116 estudios incluidos y 50 estudios focales, pero el argumento no descansa en el número por sí mismo. Descansa en la operación conceptual que la revisión realiza sobre el

corpus: definir una unidad de comparación, explicitar sus mecanismos, declarar sus límites y convertir esos límites en una agenda acumulativa. Ahí está la diferencia entre una revisión que enumera literatura y una revisión que aporta una posición al campo.

La regla editorial derivada es exigente: cada tabla, figura o anexo debe modificar la comprensión del argumento. Si un elemento visual solo ilustra, se mueve al suplemento; si muestra una relación, una frontera inferencial o una decisión de síntesis, pertenece al cuerpo del artículo. La autoridad de una revisión sistemática no aumenta por acumular material, sino por hacer visible la estructura intelectual que convierte evidencia dispersa en conocimiento discutible.

Anexo A. Fichas analíticas del corpus final incluido

Este anexo constituye la aportación analítica por estudio del subconjunto focal usado en la síntesis focal del artículo. Cada ficha combina una lectura interpretativa breve, una tabla compacta homogénea y una línea comparativa final para facilitar lectura académica, contraste metodológico y reutilización del corpus.

El bloque principal reúne 50 estudios con extracción consolidada. En esta revisión no hay estudios focales con extracción provisional dentro del bloque principal.

Estudio 1. MELON: Provable Defense Against Indirect Prompt Injection Attacks in AI Agents

MELON: Provable Defense Against Indirect Prompt Injection Attacks in AI Agents entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Re-ejecución enmascarada (masked re-execution) y comparación de herramientas (tool comparison) frente a Ataques de inyección indirecta de prompts (IPI) donde tareas maliciosas se incrustan en información recuperada por herramientas, con aplicación en Detección en tiempo de ejecución del agente, re-ejecutando la trayectoria con un prompt de usuario enmascarado. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: MELON, basado en re-ejecución con prompt enmascarado, detecta ataques de inyección indirecta de prompts comparando la similitud de acciones.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Ataques de inyección indirecta de prompts (IPI) donde tareas maliciosas se incrustan en información recuperada por herramientas mediante Re-ejecución enmascarada (masked re-execution) y comparación de herramientas (tool comparison), aplicado en Detección en tiempo de ejecución del agente, re-ejecutando la trayectoria con un prompt de usuario enmascarado; la capacidad del atacante se clasifica como no reportado. El comparador principal es Defensas SOTA contra IPI, defensa de aumento de prompts (MELON-Aug) y detector basado en LLM, entre otros. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como Preservación de utilidad superior a defensas SOTA, sin cifra concreta en el resumen y el sobrecoste como no reportado; no reportado. La evidencia de robustez es MELON supera a defensas SOTA en AgentDojo tanto en prevención de ataques como en preservación de utilidad; el estudio de ablación valida los diseños clave; MELON-Aug mejora aún más el rendimiento. y el fallo residual recuperable es Falsos positivos y falsos negativos; se incorporan tres diseños clave para reducirlos.. Para la pregunta de investigación, esto importa así: MELON, basado en re-ejecución con prompt enmascarado, detecta ataques de inyección indirecta de prompts comparando la similitud de acciones. La confianza de extracción es alta y permite integrar el estudio en la comparación central con una base empírica suficiente.

- Autores: Kaijie Zhu; Xianjun Yang; Jindong Wang; Wenbo Guo; William Yang Wang
- Año: 2025
- DOI: 10.48550/arxiv.2502.05174

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	MELON, GPT-4o, GPT-4 y ChatGPT, entre otros (n=11)
Harness o defensa	MELON
Arquitectura de control	Re-ejecución enmascarada (masked re-execution) y comparación de herramientas (tool comparison)
Punto de aplicación	Detección en tiempo de ejecución del agente, re-ejecutando la trayectoria con un prompt de usuario enmascarado
Modelo de amenaza	Ataques de inyección indirecta de prompts (IPI) donde tareas maliciosas se incrustan en información recuperada por herramientas
Tipo de ataque	Inyección indirecta de prompts (IPI) en contenido recuperado por herramientas
Adaptatividad del atacante	no reportado
Entorno o benchmark	Benchmark AgentDojo (entorno de agente con herramientas y tareas)
Baseline o comparador	Defensas SOTA contra IPI, defensa de aumento de prompts (MELON-Aug) y detector basado en LLM, entre otros
Métricas de seguridad	Attack Success Rate (ASR), tasa de falsos positivos (FPR), tasa de falsos negativos (FNR), utilidad preservada
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	Preservación de utilidad superior a defensas SOTA, sin cifra concreta en el resumen
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	MELON supera a defensas SOTA en AgentDojo tanto en prevención de ataques como en preservación de utilidad; el estudio de ablación valida los diseños clave; MELON-Aug mejora aún más el rendimiento.
Modos de fallo	Falsos positivos y falsos negativos; se incorporan tres diseños clave para reducirlos.
Código o artefactos	Disponible en https://github.com/kaijiezh11/MELON
Conclusión defensiva	MELON es una defensa contra IPI que identifica ataques comparando la trayectoria original con una re-ejecución enmascarada del prompt de usuario. Frente a defensas SOTA, reduce el ASR mientras preserva la utilidad, y su combinación con aumento de prompts (MELON-Aug) mejora el rendimiento. La amenaza son inyecciones indirectas en datos de herramientas; no se modela atacante adaptativo; el coste es bajo al ser training-free. La evidencia se basa en AgentDojo, aunque no se reportan valores numéricos en el resumen disponible.
Confianza de extracción	85

- Hallazgos clave: MELON, basado en re-ejecución con prompt enmascarado, detecta ataques de inyección indirecta de prompts comparando la similitud de acciones. En el benchmark AgentDojo supera a defensas SOTA en prevención de ataques y preservación de utilidad, y su combinación con aumento de prompts (MELON-Aug) mejora aún más el rendimiento.
- Evidencia de apoyo: Extensive evaluation on the IPI benchmark AgentDojo demonstrates that MELON outperforms SOTA defenses in both attack prevention and utility preservation.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 2. Constitutional Classifiers: Defending against Universal Jailbreaks across Thousands of Hours of Red Teaming

Constitutional Classifiers: Defending against Universal Jailbreaks across Thousands of Hours of Red Teaming entra en la revisión porque aporta evidencia trazable en la familia jailbreak y cumplimiento de políticas. Evalúa Clasificador basado en LLM entrenado con datos sintéticos generados mediante una constitución (reglas en lenguaje natural) que especifica contenido permitido y restringido. frente a Atacante que busca jailbreaks universales (prompts que eluden sistemáticamente las salvaguardas) para extraer información detallada sobre procesos dañinos (ej. fabricación de sustancias ilegales)., con aplicación en Post-procesamiento de la salida del LLM (clasificador que filtra respuestas). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante que busca jailbreaks universales (prompts que eluden sistemáticamente las salvaguardas) para extraer información detallada sobre procesos dañinos (ej. fabricación de sustancias ilegales). mediante Clasificador basado en LLM entrenado con datos sintéticos generados mediante una constitución (reglas en lenguaje natural) que especifica contenido permitido y restringido., aplicado en Post-procesamiento de la salida del LLM (clasificador que filtra respuestas); la capacidad del atacante se clasifica como Alta: red teamers humanos adaptan sus ataques durante miles de horas; se menciona que ningún red teamer encontró un jailbreak universal exitoso contra el clasificador temprano.. El comparador principal es Modelo helpful-only (sin salvaguarda), early classifier y enhanced classifier. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Los clasificadores mejorados demostraron defensa robusta contra jailbreaks específicos de dominio no vistos en evaluaciones automatizadas. y el fallo residual recuperable es No se reportan fallos específicos; se indica que ningún red teamer encontró un jailbreak universal exitoso.. Para la pregunta de investigación, esto importa así: el estudio aporta evidencia sustantiva que ayuda a delimitar mecanismo, contexto y alcance inferencial. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Mrinank Sharma; Meg Tong; Jesse Mu; Jerry Wei; Jorrit Kruthoff; Scott Goodfriend; Euan Ong; Alwin Peng; Raj Agarwal; Cem Anil; Amanda Askell; Nathan Bailey; Joe Benton; Emma Bluemke; Samuel R. Bowman; Eric Christiansen; Hoagy Cunningham; Andy Dau; Anjali Gopal; Rob Gilson; Logan Graham; Logan Howard; Nimit Kalra; Taesung Lee; Kevin Lin; Peter Lofgren; Francesco Mosconi; Clare O'Hara; Catherine Olsson; Linda Petrini; Samir Rajani; Nikhil Saxena; Alex Silverstein; Tanya Singh; Theodore Sumers; Leonard Tang; Kevin K. Troy; Constantin Weisser; Ruiqi Zhong; Giulio Zhou; Jan Leike; Jared Kaplan; Ethan Perez
- Año: 2025
- DOI: 10.48550/arxiv.2501.18837

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	Constitutional Classifiers (early classifier, enhanced classifiers), modelo base no especificado (presumiblemente un LLM de Anthropic) y modelo (n=3)
Harness o defensa	Constitutional Classifiers
Arquitectura de control	Clasificador basado en LLM entrenado con datos sintéticos generados mediante una constitución (reglas en lenguaje natural) que especifica contenido permitido y restringido.
Punto de aplicación	Post-procesamiento de la salida del LLM (clasificador que filtra respuestas)
Modelo de amenaza	Atacante que busca jailbreaks universales (prompts que eluden sistemáticamente las salvaguardas) para extraer información detallada sobre procesos dañinos (ej. fabricación de sustancias ilegales).
Tipo de ataque	Jailbreak universal (prompting adversario)
Adaptatividad del atacante	Alta: red teamers humanos adaptan sus ataques durante miles de horas; se menciona que ningún red teamer encontró un jailbreak universal exitoso contra el clasificador temprano.
Entorno o benchmark	Red teaming humano (más de 3.000 horas) y evaluaciones automatizadas con jailbreaks específicos de dominio no vistos.
Baseline o comparador	Modelo helpful-only (sin salvaguarda), early classifier y enhanced classifier
Métricas de seguridad	Tasa de éxito de ataque (ASR); tasa de rechazos en tráfico de producción; sobrecarga de inferencia no reportado
Tasa de éxito del ataque	We prioritized robustness in the classifier design and thresholds, which led to a substantial false-positive rate, with approximately 44% of Claude.ai traffic being refused.
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Los clasificadores mejorados demostraron defensa robusta contra jailbreaks específicos de dominio no vistos en evaluaciones automatizadas.
Modos de fallo	No se reportan fallos específicos; se indica que ningún red teamer encontró un jailbreak universal exitoso.
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	Los Constitutional Classifiers defienden eficazmente contra jailbreaks universales en un entorno de red teaming extenso (más de 3.000 horas) sin que se encuentre un ataque universal exitoso, con un impacto mínimo en utilidad (0,38% más rechazos) y una sobrecarga de inferencia del 23,7%, demostrando que es factible mantener la viabilidad de despliegue.

Campo	Valor
Confianza de extracción	95

- Hallazgos clave: Los Constitutional Classifiers, entrenados con datos sintéticos y una constitución, lograron una defensa completa contra jailbreaks universales en más de 3.000 horas de red teaming, con un impacto mínimo en la utilidad (0,38% más rechazos) y una sobrecarga de inferencia del 23,7%, demostrando viabilidad práctica.
- Evidencia de apoyo: In over 3,000 estimated hours of red teaming, no red teamer found a universal jailbreak that could extract information from an early classifier-guarded LLM at a similar level of detail to an unguarded model across most target queries.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre jailbreak y cumplimiento de políticas y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 3. Securing LLM-Agent Long-Term Memory Against Poisoning: Non-Malleable, Origin-Bound Authority with Machine-Checked Guarantees

Securing LLM-Agent Long-Term Memory Against Poisoning: Non-Malleable, Origin-Bound Authority with Machine-Checked Guarantees entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Control de flujo de información no maleable (non-malleable IFC) con vinculación de origen en escritura y elevación controlada por corroboración resistente a Sybil. frente a Adversario que puede almacenar contenido no confiable en una sesión para dirigir acciones consecuentes en sesiones futuras, con capacidad de blanquear origen a través de resumen del agente, eco de herramienta confiable o corroboración fabricada., con aplicación en Escritura de memoria (write-time origin binding) y recuperación (corroboration-gated elevation). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: TMA-NM logra un 0% de éxito de ataque en todos los modelos y canales de blanqueo, mientras que las defensas existentes fallan hasta en un 68% de los casos, manteniendo la utilidad legítima completa.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Adversario que puede almacenar contenido no confiable en una sesión para dirigir acciones consecuentes en sesiones futuras, con capacidad de blanquear origen a través de resumen del agente, eco de herramienta confiable o corroboración fabricada. mediante Control de flujo de información no maleable (non-malleable IFC) con vinculación de origen en escritura y elevación controlada por corroboración resistente a Sybil., aplicado en Escritura de memoria (write-time origin binding) y recuperación (corroboration-gated elevation); la capacidad del atacante se clasifica como El atacante puede adaptar el contenido envenenado para evadir defensas basadas en contenido o linaje; se evalúa en múltiples canales de blanqueo.. El comparador principal es Defensas existentes basadas en contenido (detección/puntuación de confianza) y linaje (derivación) y configuración sin defensa (none).. La eficacia se reporta como TMA-NM is the only one of the five defense classes at 0% ASR on both the direct attack and laundering, at 100% legit-utility, identical to the undefended agent., el efecto sobre uso legítimo como TMA-NM mantiene utilidad legítima completa (full legitimate utility); las defensas existentes muestran degradación de utilidad al ajustar umbrales. y el sobre-coste como no reportado; no reportado. La evidencia de robustez es TMA-NM alcanza 0% ASR en todos los modelos y canales, con utilidad completa; verificación formal con TLA+; teorema de separación demostrado. y el fallo residual recuperable es Las defensas basadas en contenido y linaje fallan bajo blanqueo (hasta 68% ASR); ningún ajuste de umbral cierra la brecha estructural.. Para la pregunta de investigación, esto importa así: TMA-NM logra un 0% de éxito de ataque en todos los modelos y canales de blanqueo, mientras que las defensas existentes fallan hasta en un 68% de los casos, manteniendo la utilidad legítima completa. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Yedidel Louck
- Año: 2026
- DOI: 10.48550/arxiv.2606.24322

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	gpt-5-chat, gpt-4o-mini, claude-opus-4.1 y claude-sonnet-4.5, entre otros (n=8)
Harness o defensa	TMA-NM (Tamper-evident Memory Authority, Non-Malleable)
Arquitectura de control	Control de flujo de información no maleable (non-malleable IFC) con vinculación de origen en escritura y elevación controlada por corroboración resistente a Sybil.
Punto de aplicación	Escritura de memoria (write-time origin binding) y recuperación (corroboration-gated elevation)
Modelo de amenaza	Adversario que puede almacenar contenido no confiable en una sesión para dirigir acciones consecuentes en sesiones futuras, con capacidad de blanquear origen a través de resumen del agente, eco de herramienta confiable o corroboración fabricada.
Tipo de ataque	Envenenamiento de memoria (memory poisoning) directo y mediante blanqueo (laundering) a través de tres canales: resumen del agente, eco de herramienta confiable y corroboración fabricada.
Adaptatividad del atacante	El atacante puede adaptar el contenido envenenado para evadir defensas basadas en contenido o linaje; se evalúa en múltiples canales de blanqueo.
Entorno o benchmark	Benchmark controlado con 8 modelos, 4 canales de ataque, 24 repeticiones por celda; se mide tasa de éxito de ataque (ASR) y utilidad legítima.
Baseline o comparador	Defensas existentes basadas en contenido (detección/puntuación de confianza) y linaje (derivación) y configuración sin defensa (none).
Métricas de seguridad	Attack Success Rate (ASR) para ataques consecuentes; utilidad legítima (legitimate utility); intervalos de confianza Wilson (95%)
Tasa de éxito del ataque	TMA-NM is the only one of the five defense classes at 0% ASR on both the direct attack and laundering, at 100% legit-utility, identical to the undefended agent.
Falsos positivos	no reportado
Impacto en utilidad	TMA-NM mantiene utilidad legítima completa (full legitimate utility); las defensas existentes muestran degradación de utilidad al ajustar umbrales.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	TMA-NM alcanza 0% ASR en todos los modelos y canales, con utilidad completa; verificación formal con TLA+; teorema de separación demostrado.

Campo	Valor
Modos de fallo	Las defensas basadas en contenido y linaje fallan bajo blanqueo (hasta 68% ASR); ningún ajuste de umbral cierra la brecha estructural.
Código o artefactos	Sí, se liberan benchmark, harness y modelos TLA+ verificados por máquina.
Conclusión defensiva	Las defensas existentes basadas en contenido o linaje son insuficientes frente a ataques de blanqueo en memoria de agentes LLM (hasta 68% ASR). TMA-NM, basado en control de flujo de información no maleable con vinculación de origen en escritura y elevación controlada por corroboración resistente a Sybil, alcanza 0% ASR en todos los modelos y canales evaluados, manteniendo utilidad completa. La solución es robusta y está formalmente verificada.
Confianza de extracción	95

- Hallazgos clave: TMA-NM logra un 0% de éxito de ataque en todos los modelos y canales de blanqueo, mientras que las defensas existentes fallan hasta en un 68% de los casos, manteniendo la utilidad legítima completa.
- Evidencia de apoyo: TMA-NM reaches 0% attack success on both direct and laundering attacks across all models and channels, at full legitimate utility.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 4. Cognitive Control Architecture (CCA): A Lifecycle Supervision Framework for Robustly Aligned AI Agents

Cognitive Control Architecture (CCA): A Lifecycle Supervision Framework for Robustly Aligned AI Agents entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Doble capa: (i) grafo de intenciones pre-generado para integridad de flujo de control/datos; (ii) Tiered Adjudicator que detecta desviaciones y razona con puntuación multidimensional. frente a Atacante que inyecta instrucciones maliciosas en fuentes de información externas consultadas por el agente (indirect prompt injection)., con aplicación en Durante la planificación y ejecución de acciones del agente, antes de invocar herramientas.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: CCA reduce el ASR por debajo del 1% en AgentDojo frente a inyección indirecta de prompts, manteniendo una utilidad benigna del 87.63% y una utilidad bajo ataque del $86.47 \pm 0.82\%$.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante que inyecta instrucciones maliciosas en fuentes de información externas consultadas por el agente (indirect prompt injection). mediante Doble capa: (i) grafo de intenciones pre-generado para integridad de flujo de control/datos; (ii) Tiered Adjudicator que detecta desviaciones y razona con puntuación multidimensional., aplicado en Durante la planificación y ejecución de acciones del agente, antes de invocar herramientas.; la capacidad del atacante se clasifica como ataque estático. El comparador principal es No Defense, DeBERTa classifier y Spotlight, entre otros. La eficacia se reporta como ASR medio $< 1\%$ ($0.72\% \pm 0.14\%$ en ataque Important Messages), el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es ASR bajo consistente en 5 ejecuciones; actualización dinámica del grafo: ASR estático 0.63% frente a $0.72 \pm 0.14\%$ del modelo completo; UA baja de 86.47% a

84.19% sin actualizaciones. y el fallo residual recuperable es no reportado. Para la pregunta de investigación, esto importa así: CCA reduce el ASR por debajo del 1% en AgentDojo frente a inyección indirecta de prompts, manteniendo una utilidad benigna del 87.63% y una utilidad bajo ataque del $86.47 \pm 0.82\%$. La confianza de extracción es alta y permite integrar el estudio en la comparación central con una base empírica suficiente.

- Autores: Zhibo Liang; Tianze Hu; Zaiye Chen; Mingjie Tang
- Año: 2025
- DOI: 10.48550/arxiv.2512.06716

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	CCA, No Defense, DeBERTa y Spotlight, entre otros (n=6)
Harness o defensa	Cognitive Control Architecture (CCA)
Arquitectura de control	Doble capa: (i) grafo de intenciones pre-generado para integridad de flujo de control/datos; (ii) Tiered Adjudicator que detecta desviaciones y razona con puntuación multidimensional.
Punto de aplicación	Durante la planificación y ejecución de acciones del agente, antes de invocar herramientas.
Modelo de amenaza	Atacante que inyecta instrucciones maliciosas en fuentes de información externas consultadas por el agente (indirect prompt injection).
Tipo de ataque	Indirect Prompt Injection (variantes: Direct, Ignore Previous, System Message, Important Messages)
Adaptatividad del atacante	ataque estático
Entorno o benchmark	Evaluación offline sobre el benchmark AgentDojo; tareas fijas y ataques predefinidos.
Baseline o comparador	No Defense, DeBERTa classifier y Spotlight, entre otros
Métricas de seguridad	ASR (Attack Success Rate), UA (Utility Under Attack), BU (Benign Utility), Efficiency Overhead (llamadas API y latencia)
Tasa de éxito del ataque	ASR medio $< 1\%$ ($0.72\% \pm 0.14\%$ en ataque Important Messages)
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	ASR bajo consistente en 5 ejecuciones; actualización dinámica del grafo: ASR estático 0.63% frente a $0.72 \pm 0.14\%$ del modelo completo; UA baja de 86.47% a 84.19% sin actualizaciones.
Modos de fallo	no reportado
Código o artefactos	no reportado

Campo	Valor
Conclusión defensiva	CCA logra un ASR muy bajo (por debajo del 1%) en AgentDojo frente a IPI, superando a las baselines (DeBERTa, Spotlight, Repeat Prompt, MELON) y manteniendo utilidad alta (UA y BU). No se reportan tasas de falsos positivos, coste de latencia ni adaptatividad del atacante; la evidencia se limita a un único benchmark con ataques conocidos, por lo que la superioridad no debe generalizarse sin más evaluación.
Confianza de extracción	88

- Hallazgos clave: CCA reduce el ASR por debajo del 1% en AgentDojo frente a inyección indirecta de prompts, manteniendo una utilidad benigna del 87.63% y una utilidad bajo ataque del $86.47 \pm 0.82\%$. La ablación muestra que la actualización dinámica del grafo preserva la seguridad y mejora la utilidad.
- Evidencia de apoyo: CCA maintains a very low ASR (below 1%) with small variance, while preserving high benign utility.
- Localización de la evidencia: p. 22 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 5. Geometry-Guided Adversarial Prompt Detection via Curvature and Local Intrinsic Dimension

Geometry-Guided Adversarial Prompt Detection via Curvature and Local Intrinsic Dimension entra en la revisión porque aporta evidencia trazable en la familia jailbreak y cumplimiento de políticas. Evalúa Módulo de detección externo que analiza propiedades geométricas de los embeddings del prompt; no modifica el LLM subyacente. frente a Atacante que genera prompts adversariales diseñados para jailbreak del LLM y producir salidas dañinas., con aplicación en Antes de que el prompt sea procesado por el LLM (pre-detección). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: CurvaLID demuestra que los prompts adversariales tienen firmas geométricas distinguibles, alcanzando una clasificación casi perfecta y un mejor rendimiento que detectores como Llama Guard y Perplexity Filtering en la reducción de ASR.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en Análisis geométrico de prompts mediante curvatura (ecuación de Whewell) y Local Intrinsic Dimensionality (LID); clasificación supervisada binaria.. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante que genera prompts adversariales diseñados para jailbreak del LLM y producir salidas dañinas. mediante Módulo de detección externo que analiza propiedades geométricas de los embeddings del prompt; no modifica el LLM subyacente., aplicado en Antes de que el prompt sea procesado por el LLM (pre-detección); la capacidad del atacante se clasifica como ataque adaptativo. El comparador principal es Circuit Breakers, Llama Guard 3 y Perplexity Filtering, entre otros. La eficacia se reporta como

- CurvaLID successfully detects about 99% of adversarial prompts, outperforming state-of-the-art defences by over 10% in attack success rate reduction across multiple LLMs and adversarial attacks., el efecto sobre uso legítimo como For Vicuna-7B, CurvaLID reduced harmful acceptance by up to 30%, with only a modest increase (about 10%) in benign rejections, suggesting it can enhance safety without substantially impacting utility. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Logra una clasificación casi perfecta y supera a los detectores SOTA en varios LLMs y familias de ataque. y el fallo residual recuperable es no reportado. Para la pregunta de investigación, esto importa así: CurvaLID demuestra que los prompts adversariales tienen firmas geométricas distinguibles, alcanzando una clasificación casi perfecta y un mejor rendimiento que detectores como Llama Guard y Perplexity Filtering en la reducción de ASR. La

confianza de extracción es alta y permite integrar el estudio en la comparación central con una base empírica suficiente.

- Autores: Canaan Yung; Hanxun Huang; Christopher Leckie; Sarah Erfani
- Año: 2025
- DOI: 10.48550/arxiv.2503.03502

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	LLaMA-3-8B y Llama Guard 3 (otros LLMs no especificados en el digest) (n=10)
Harness o defensa	CurvaLID
Arquitectura de control	Módulo de detección externo que analiza propiedades geométricas de los embeddings del prompt; no modifica el LLM subyacente.
Punto de aplicación	Antes de que el prompt sea procesado por el LLM (pre-detección)
Modelo de amenaza	Atacante que genera prompts adversariales diseñados para jailbreak del LLM y producir salidas dañinas.
Tipo de ataque	Jailbreak de prompt; ataques adversariales (GCG, PAIR, DAN, AutoDAN, AmpleGCG, SAP, RandomSearch, Persuasive Attack)
Adaptatividad del atacante	ataque adaptativo
Entorno o benchmark	Evaluación offline de clasificación binaria sobre prompts adversariales y benignos en múltiples LLMs; se reporta ASR por ataque y modelo.
Baseline o comparador	Circuit Breakers, Llama Guard 3 y Perplexity Filtering, entre otros
Métricas de seguridad	ASR (Attack Success Rate); precisión de clasificación (near-perfect classification)
Tasa de éxito del ataque	• CurvaLID successfully detects about 99% of adversarial prompts, outperforming state-of-the-art defences by over 10% in attack success rate reduction across multiple LLMs and adversarial attacks.
Falsos positivos	no reportado
Impacto en utilidad	For Vicuna-7B, CurvaLID reduced harmful acceptance by up to 30%, with only a modest increase (about 10%) in benign rejections, suggesting it can enhance safety without substantially impacting utility.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Logra una clasificación casi perfecta y supera a los detectores SOTA en varios LLMs y familias de ataque.
Modos de fallo	no reportado
Código o artefactos	código o artefacto reportado como disponible

Campo	Valor
Conclusión defensiva	CurvaLID es un detector eficiente y agnóstico al modelo que identifica prompts adversariales con alta precisión, superando a Circuit Breakers, Llama Guard 3 y Perplexity Filtering en la reducción de ASR; sin embargo, no se reportan tasas de falsos positivos, ni sobrecoste de inferencia, ni impacto en utilidad, por lo que su ventaja debe interpretarse con cautela más allá del benchmark evaluado.
Confianza de extracción	80

- Hallazgos clave: CurvaLID demuestra que los prompts adversariales tienen firmas geométricas distinguibles, alcanzando una clasificación casi perfecta y un mejor rendimiento que detectores como Llama Guard y Perplexity Filtering en la reducción de ASR.
- Evidencia de apoyo: Our findings show that adversarial prompts exhibit distinct geometric signatures from benign prompts, enabling CurvaLID to achieve near-perfect classification and outperform state-of-the-art detectors in adversarial prompt detection.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre jailbreak y cumplimiento de políticas y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 6. Taxonomy, Evaluation and Exploitation of IPI-Centric LLM Agent Defense Frameworks

Taxonomy, Evaluation and Exploitation of IPI-Centric LLM Agent Defense Frameworks entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Agente LLM con llamada a funciones (sin defensa específica) frente a Atacante externo que controla datos externos recibidos por el agente para inyectar prompts maliciosos que secuestran llamadas a herramientas, con aplicación en Antes de la ejecución de la herramienta (pre-llamada); durante la ejecución; post-llamada. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Se presenta la primera taxonomía completa de marcos de defensa contra inyección indirecta de prompts en agentes LLM, evaluando 11 marcos en tres benchmarks.

Desde el punto de vista del diseño, se trata de un estudio de revisión apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc sobre benchmarks con ataques adaptativos diseñados a partir de análisis de fallos defensivos. Protege frente a Atacante externo que controla datos externos recibidos por el agente para inyectar prompts maliciosos que secuestran llamadas a herramientas mediante Agente LLM con llamada a funciones (sin defensa específica), aplicado en Antes de la ejecución de la herramienta (pre-llamada); durante la ejecución; post-llamada; la capacidad del atacante se clasifica como Adaptativo: se diseñan tres nuevos ataques adaptativos basados en el análisis de fallos defensivos. El comparador principal es Sin defensa (baseline), LlamaFirewall y Tool Filter, entre otros. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como Meta SecAlign [17] is the first open-source LLM with a built-in model-level defense based on an improved SecAlign method that achieves state-of-the-art robustness against prompt injection attacks and comparable utility to closed-source commercial LLMs. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Los marcos basados en políticas y diseño de sistema logran el ASR más bajo; los basados en ingeniería de prompts, verificación en tiempo real y detección muestran ASR ligeramente más alto pero aún bajo; se identifican seis causas raíz de evasión y el fallo residual recuperable es Seis causas raíz de evasión de defensas identificadas (no detalladas en el digest). Para la pregunta de investigación, esto importa así: Se presenta la primera taxonomía completa de marcos de defensa contra inyección indirecta de prompts en agentes LLM, evaluando 11 marcos en tres benchmarks. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Zimo Ji; Xunguang Wang; Zongjie Li; Pingchuan Ma; Yudong Gao; Daoyuan Wu; Xincheng Yan; Tian Tian; Shuai Wang
- Año: 2025
- DOI: 10.48550/arxiv.2511.15203

Campo	Valor
Tipo de trabajo	Revisión
Diseño	Evaluación experimental post hoc sobre benchmarks con ataques adaptativos diseñados a partir de análisis de fallos defensivos
Modelos o sistemas protegidos	GPT-4o, DeepSeek-V3, LlamaFirewall y Tool Filter, entre otros (n=11)
Harness o defensa	Taxonomía y evaluación de marcos de defensa IPI (SoK)
Arquitectura de control	Agente LLM con llamada a funciones (sin defensa específica)
Punto de aplicación	Antes de la ejecución de la herramienta (pre-llamada); durante la ejecución; post-llamada
Modelo de amenaza	Atacante externo que controla datos externos recibidos por el agente para inyectar prompts maliciosos que secuestran llamadas a herramientas
Tipo de ataque	Indirect Prompt Injection (IPI)
Adaptatividad del atacante	Adaptativo: se diseñan tres nuevos ataques adaptativos basados en el análisis de fallos defensivos
Entorno o benchmark	Benchmarks estandarizados (AgentDojo, ASB, InjecAgent) con GPT-4o y DeepSeek-V3 como modelos de agente
Baseline o comparador	Sin defensa (baseline), LlamaFirewall y Tool Filter, entre otros
Métricas de seguridad	Attack Success Rate (ASR); tasa de falsos positivos (no reportada explícitamente); impacto en utilidad (no reportado explícitamente)
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	Meta SecAlign [17] is the first open-source LLM with a built-in model-level defense based on an improved SecAlign method that achieves state-of-the-art robustness against prompt injection attacks and comparable utility to closed-source commercial LLMs.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Los marcos basados en políticas y diseño de sistema logran el ASR más bajo; los basados en ingeniería de prompts, verificación en tiempo real y detección muestran ASR ligeramente más alto pero aún bajo; se identifican seis causas raíz de evasión
Modos de fallo	Seis causas raíz de evasión de defensas identificadas (no detalladas en el digest)
Código o artefactos	código o artefacto reportado como disponible

Campo	Valor
Conclusión defensiva	Los marcos de defensa IPI existentes reducen significativamente el ASR frente a ataques baseline, pero siguen siendo vulnerables a ataques adaptativos diseñados explotando las causas raíz de evasión. No se declara un marco superior de forma absoluta; la efectividad varía según el benchmark y el modelo de agente. Se necesita un desarrollo más robusto y usable.
Confianza de extracción	95

- Hallazgos clave: Se presenta la primera taxonomía completa de marcos de defensa contra inyección indirecta de prompts en agentes LLM, evaluando 11 marcos en tres benchmarks. Los ataques adaptativos diseñados a partir de seis causas raíz de evasión mejoran significativamente las tasas de éxito, demostrando que las defensas actuales son aún frágiles.
- Evidencia de apoyo: In this Systematization of Knowledge (SoK), we present the first comprehensive analysis of IPI-centric defense frameworks.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 7. AgentSentry: Mitigating Indirect Prompt Injection in LLM Agents via Temporal Causal Diagnostics and Context Purification

AgentSentry: Mitigating Indirect Prompt Injection in LLM Agents via Temporal Causal Diagnostics and Context Purification entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Inferencia con diagnóstico causal temporal: re-ejecuciones contrafactuales en límites de retorno de herramientas y purificación de contexto guiada causalmente frente a Indirect prompt injection (IPI) donde el contexto controlado por el atacante se incrusta en salidas de herramientas o contenido recuperado, con aplicación en Límites de retorno de herramientas. Metodológicamente se articula como Evaluación experimental post hoc sobre el benchmark AgentDojo con comparación de defensas y múltiples LLMs de caja negra. Su aportación central es: AgentSentry elimina los ataques de inyección indirecta de prompts en el benchmark AgentDojo y mantiene una utilidad alta bajo ataque (UA media del 74.55%), superando a las líneas base sin degradar el rendimiento benigno.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo cuantitativo apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc sobre el benchmark AgentDojo con comparación de defensas y múltiples LLMs de caja negra. Protege frente a Indirect prompt injection (IPI) donde el contexto controlado por el atacante se incrusta en salidas de herramientas o contenido recuperado mediante Inferencia con diagnóstico causal temporal: re-ejecuciones contrafactuales en límites de retorno de herramientas y purificación de contexto guiada causalmente, aplicado en Límites de retorno de herramientas; la capacidad del atacante se clasifica como No especificado; se evalúan tres familias de ataque IPI predefinidas. El comparador principal es no reportado (se comparan varias defensas existentes). La eficacia se reporta como Our results demonstrate that AgentSentry successfully defends against all evaluated injection attacks, achieving an Attack Success Rate (ASR) of 0% while maintaining strong Utility Under Attack (UA)., el efecto sobre uso legítimo como AgentSentry eliminates successful attacks and maintains strong utility under attack, achieving an average Utility Under Attack (UA) of 74.55 %, improving UA by 20.8 to 33.6 percentage points over the strongest baselines without degrading benign performance. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Elimina ataques exitosos, mantiene utilidad bajo ataque y no degrada el rendimiento benigno y el fallo residual recuperable es We evaluate security and task performance using four metrics summarized in Table 1: Attack Success

Rate (ASR, ↓), Utility under Attack (UA, ↑), Clean Utility (CU, ↑), and False Positive Rate (FPR, ↓).. Para la pregunta de investigación, esto importa así: AgentSentry elimina los ataques de inyección indirecta de prompts en el benchmark AgentDojo y mantiene una utilidad alta bajo ataque (UA media del 74.55%), superando a las líneas base sin degradar el rendimiento benigno. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Tian Zhang; Yiwei Xu; Juan Wang; Keyan Guo; Xiaoyang Xu; Bowen Xiao; Quanlong Guan; Jinlin Fan; Jiawei Liu; Zhiquan Liu; Hongxin Hu
- Año: 2026
- DOI: 10.48550/arxiv.2602.22724

Campo	Valor
Tipo de trabajo	Empírico
Diseño	Evaluación experimental post hoc sobre el benchmark AgentDojo con comparación de defensas y múltiples LLMs de caja negra
Modelos o sistemas protegidos	no reportado (se mencionan múltiples LLMs de caja negra) (n=0)
Harness o defensa	AgentSentry
Arquitectura de control	Inferencia con diagnóstico causal temporal: re-ejecuciones contrafactuales en límites de retorno de herramientas y purificación de contexto guiada causalmente
Punto de aplicación	Límites de retorno de herramientas
Modelo de amenaza	Indirect prompt injection (IPI) donde el contexto controlado por el atacante se incrusta en salidas de herramientas o contenido recuperado
Tipo de ataque	Indirect prompt injection (IPI)
Adaptatividad del atacante	No especificado; se evalúan tres familias de ataque IPI predefinidas
Entorno o benchmark	Benchmark estandarizado AgentDojo; entornos de tareas con herramientas
Baseline o comparador	no reportado (se comparan varias defensas existentes)
Métricas de seguridad	Utility Under Attack (UA); tasa de éxito de ataque; rendimiento benigno
Tasa de éxito del ataque	Our results demonstrate that AgentSentry successfully defends against all evaluated injection attacks, achieving an Attack Success Rate (ASR) of 0% while maintaining strong Utility Under Attack (UA).
Falsos positivos	no reportado
Impacto en utilidad	AgentSentry eliminates successful attacks and maintains strong utility under attack, achieving an average Utility Under Attack (UA) of 74.55 %, improving UA by 20.8 to 33.6 percentage points over the strongest baselines without degrading benign performance.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Elimina ataques exitosos, mantiene utilidad bajo ataque y no degrada el rendimiento benigno

Campo	Valor
Modos de fallo	We evaluate security and task performance using four metrics summarized in Table 1: Attack Success Rate (ASR, ↓), Utility under Attack (UA, ↑), Clean Utility (CU, ↑), and False Positive Rate (FPR, ↓).
Código o artefactos	no reportado
Conclusión defensiva	AgentSentry, una defensa en tiempo de inferencia, reduce la tasa de éxito de IPI a cero en el benchmark AgentDojo, manteniendo una utilidad bajo ataque del 74.55% y superando a las mejores líneas base en 20.8-33.6 puntos porcentuales, sin sobre coste reportado ni degradación benigna; sin embargo, la adaptatividad del atacante y el sobre coste en latencia/coste no se reportan explícitamente.
Confianza de extracción	90

- Hallazgos clave: AgentSentry elimina los ataques de inyección indirecta de prompts en el benchmark AgentDojo y mantiene una utilidad alta bajo ataque (UA media del 74.55%), superando a las líneas base sin degradar el rendimiento benigno.
- Evidencia de apoyo: AgentSentry eliminates successful attacks and maintains strong utility under attack, achieving an average Utility Under Attack (UA) of 74.55%
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 8. LivePI: More Realistic Benchmarking of Agents Against Indirect Prompt Injection

LivePI: More Realistic Benchmarking of Agents Against Indirect Prompt Injection entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Agente OpenClaw con backbones LLM; defensa de dos capas: filtrado de instrucciones a nivel de prompt y autorización de llamadas a herramientas previa a la ejecución frente a Atacante externo que incrusta instrucciones maliciosas en contenido aparentemente benigno (correo, archivos, páginas web, repositorios, mensajes de chat grupal) que el agente procesa, con aplicación en Pre-ejecución (filtrado de prompt y autorización de tool-call); también se menciona inspección post-llamada a herramienta para actualizar estado y evidencia. Metodológicamente se articula como Evaluación experimental post hoc sobre benchmark estructurado con ataques controlados en máquina virtual real. Su aportación central es: LivePI muestra que los agentes con herramientas son vulnerables a inyección indirecta de instrucciones con tasas de éxito de ataque entre 10.7% y 29.6%, siendo el chat grupal un vector crítico con éxito uniforme.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc sobre benchmark estructurado con ataques controlados en máquina virtual real. Protege frente a Atacante externo que incrusta instrucciones maliciosas en contenido aparentemente benigno (correo, archivos, páginas web, repositorios, mensajes de chat grupal) que el agente procesa mediante Agente OpenClaw con backbones LLM; defensa de dos capas: filtrado de instrucciones a nivel de prompt y autorización de llamadas a herramientas previa a la ejecución, aplicado en Pre-ejecución (filtrado de prompt y autorización de tool-call); también se menciona inspección post-llamada a herramienta para actualizar estado y evidencia; la capacidad del atacante se clasifica como No adaptativo (ataques predefinidos, no se menciona adaptación en tiempo real). El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia

se reporta como Across GPT-5.3-Codex, Claude Opus 4.6, Gemini 3.1 Pro, Kimi K2.5, and GLM-5, total attack success rates range from 10.7% to 29.6%., el efecto sobre uso legítimo como In the GPT-5.3-Codex setting, the defense intercepts all tested malicious-goal completions in LivePI before execution while preserving benign utility on PinchBench-derived workloads. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es La defensa intercepta todas las finalizaciones de objetivos maliciosos probadas en LivePI para GPT-5.3-Codex; no se evalúa en otros modelos y el fallo residual recuperable es Inyección en chat grupal exitosa en todos los backbones; ataques con enlaces a repositorios producen fallos de alta gravedad; algunos ataques pueden eludir la defensa si no se detectan a nivel de prompt o tool-call. Para la pregunta de investigación, esto importa así: LivePI muestra que los agentes con herramientas son vulnerables a inyección indirecta de instrucciones con tasas de éxito de ataque entre 10.7% y 29.6%, siendo el chat grupal un vector crítico con éxito uniforme. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Lei Zhao; Abhay Bhaskar; Edgar Dobriban
- Año: 2026
- DOI: 10.48550/arxiv.2605.17986

Campo	Valor
Tipo de trabajo	Empírico
Diseño	Evaluación experimental post hoc sobre benchmark estructurado con ataques controlados en máquina virtual real
Modelos o sistemas protegidos	GPT-5.3-Codex, Claude Opus 4.6, Gemini 3.1 Pro y Kimi K2.5, entre otros
Harness o defensa	LivePI
Arquitectura de control	Agente OpenClaw con backbones LLM; defensa de dos capas: filtrado de instrucciones a nivel de prompt y autorización de llamadas a herramientas previa a la ejecución
Punto de aplicación	Pre-ejecución (filtrado de prompt y autorización de tool-call); también se menciona inspección post-llamada a herramienta para actualizar estado y evidencia
Modelo de amenaza	Atacante externo que incrusta instrucciones maliciosas en contenido aparentemente benigno (correo, archivos, páginas web, repositorios, mensajes de chat grupal) que el agente procesa
Tipo de ataque	Indirect prompt injection (IPI) a través de siete superficies de entrada: correo electrónico, archivos descargados, páginas web, repositorios, mensajes de chat grupal, y otras; doce familias de ataque/representación; cinco objetivos maliciosos: exfiltración de información protegida, cambios no autorizados en controles de seguridad, recuperación/ejecución de código inseguro, exfiltración de resumen de bandeja de entrada, transferencia de criptomonedas
Adaptatividad del atacante	No adaptativo (ataques predefinidos, no se menciona adaptación en tiempo real)
Entorno o benchmark	Entorno de máquina virtual real con interfaces en vivo pero controladas (correo, chat, web, archivos locales, repositorio, billetera); producción simulada

Campo	Valor
Baseline o comparador	Sin defensa (ataque directo) y defensa de dos capas (filtrado a nivel de instrucción + autorización de llamadas a herramientas previa a la
Métricas de seguridad	Attack Success Rate (ASR) total por modelo; ASR por superficie de entrada; gravedad de fallos (alta/severa); tasa de interceptación de defensa; utilidad benigna preservada (PinchBench)
Tasa de éxito del ataque	Across GPT-5.3-Codex, Claude Opus 4.6, Gemini 3.1 Pro, Kimi K2.5, and GLM-5, total attack success rates range from 10.7% to 29.6%.
Falsos positivos	no reportado
Impacto en utilidad	In the GPT-5.3-Codex setting, the defense intercepts all tested malicious-goal completions in LivePI before execution while preserving benign utility on PinchBench-derived workloads.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	La defensa intercepta todas las finalizaciones de objetivos maliciosos probadas en LivePI para GPT-5.3-Codex; no se evalúa en otros modelos
Modos de fallo	Inyección en chat grupal exitosa en todos los backbones; ataques con enlaces a repositorios producen fallos de alta gravedad; algunos ataques pueden eludir la defensa si no se detectan a nivel de prompt o tool-call
Código o artefactos	Código y benchmark disponibles en GitHub: https://github.com/leizhao7/livepi
Conclusión defensiva	LivePI revela que los agentes actuales son vulnerables a IPI con ASR entre 10.7% y 29.6%, siendo el chat grupal un vector crítico (100% éxito). La defensa de dos capas (filtrado de prompt + autorización de tool-call) es efectiva en GPT-5.3-Codex para interceptar todos los ataques probados sin degradar la utilidad en PinchBench, pero no se ha evaluado en otros modelos ni se reporta sobrecarga de latencia o coste. Se necesita más investigación para generalizar la defensa y evaluar su robustez ante ataques adaptativos.
Confianza de extracción	95

- Hallazgos clave: LivePI muestra que los agentes con herramientas son vulnerables a inyección indirecta de instrucciones con tasas de éxito de ataque entre 10.7% y 29.6%, siendo el chat grupal un vector crítico con éxito uniforme. Una defensa de dos capas (filtrado de prompt + autorización de tool-call) intercepta todos los ataques probados en GPT-5.3-Codex sin afectar la utilidad benigna.
- Evidencia de apoyo: Across GPT-5.3-Codex, Claude Opus 4.6, Gemini 3.1 Pro, Kimi K2.5, and GLM-5, total attack success rates range from 10.7% to 29.6%.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 9. Simple Prompt Injection Attacks Can Leak Personal Data Observed by LLM Agents During Task Execution

Simple Prompt Injection Attacks Can Leak Personal Data Observed by LLM Agents During Task Execution entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa agente bancario ficticio con llamadas a herramientas; arquitectura de agente tool-calling frente a atacante externo que inyecta instrucciones maliciosas en la entrada del agente para exfiltrar datos personales observados durante la tarea, con aplicación en durante la ejecución de la tarea, en el momento de la llamada a herramienta o respuesta del agente. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Los ataques simples de inyección de instrucciones logran una tasa de éxito media del 20% en 16 tareas bancarias simuladas, causando una caída de utilidad del 15-50%.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a atacante externo que inyecta instrucciones maliciosas en la entrada del agente para exfiltrar datos personales observados durante la tarea mediante agente bancario ficticio con llamadas a herramientas; arquitectura de agente tool-calling, aplicado en durante la ejecución de la tarea, en el momento de la llamada a herramienta o respuesta del agente; la capacidad del atacante se clasifica como no adaptativo (ataques predefinidos, no se adaptan dinámicamente a las defensas). El comparador principal es sin ataque (línea base de utilidad), defensas integradas en AgentDojo (4 métodos) y comparación entre modelos. La eficacia se reporta como alrededor del 20% en 16 tareas; alrededor del 15% en 48 tareas; algunas defensas reducen ASR a 0%, el efecto sobre uso legítimo como In 16 user tasks from AgentDojo, LLMs show a 15-50 percentage point drop in utility under attack, with average attack success rates (ASR) around 20 percent; some defenses reduce ASR to zero. y el sobrecoste como no reportado; coste estimado de ejecución: ~\$127.5 para modelos GPT (incluyendo todas las evaluaciones); ~\$40 para Claude 3.5 Sonnet. La evidencia de robustez es ninguna defensa integrada en AgentDojo previene completamente la fuga; las defensas pueden reducir ASR a cero en algunos casos; la fuga de contraseñas es menos probable que la de otros datos personales y el fallo residual recuperable es los agentes filtran datos personales no críticos incluso cuando evitan contraseñas; las tareas de extracción de datos y autorización son más vulnerables; la probabilidad de fuga de contraseñas aumenta cuando se solicitan detalles adicionales. Para la pregunta de investigación, esto importa así: Los ataques simples de inyección de instrucciones logran una tasa de éxito media del 20% en 16 tareas bancarias simuladas, causando una caída de utilidad del 15-50%. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Meysam Alizadeh; Zeynab Samei; Daria Stetsenko; Fabrizio Gilardi
- Año: 2025
- DOI: 10.48550/arxiv.2506.01055

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-3.5 Turbo, GPT-4o, GPT-4 y Claude 3.5 Sonnet
Harness o defensa	AgentDojo (adaptado con ataques de inyección basados en flujo de datos)
Arquitectura de control	agente bancario ficticio con llamadas a herramientas; arquitectura de agente tool-calling
Punto de aplicación	durante la ejecución de la tarea, en el momento de la llamada a herramienta o respuesta del agente
Modelo de amenaza	atacante externo que inyecta instrucciones maliciosas en la entrada del agente para exfiltrar datos personales observados durante la tarea

Campo	Valor
Tipo de ataque	inyección de instrucciones (prompt injection) basada en flujo de datos; ataques de exfiltración de datos personales
Adaptatividad del atacante	no adaptativo (ataques predefinidos, no se adaptan dinámicamente a las defensas)
Entorno o benchmark	entorno simulado de agente bancario con tareas sintéticas; evaluación offline sobre benchmark
Baseline o comparador	sin ataque (línea base de utilidad), defensas integradas en AgentDojo (4 métodos) y comparación entre modelos
Métricas de seguridad	Attack Success Rate (ASR); utilidad del agente (porcentaje de tareas completadas correctamente); tasa de fuga de contraseñas; tasa de fuga de otros datos personales
Tasa de éxito del ataque	alrededor del 20% en 16 tareas; alrededor del 15% en 48 tareas; algunas defensas reducen ASR a 0%
Falsos positivos	no reportado
Impacto en utilidad	In 16 user tasks from AgentDojo, LLMs show a 15-50 percentage point drop in utility under attack, with average attack success rates (ASR) around 20 percent; some defenses reduce ASR to zero.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	coste estimado de ejecución: ~\$127.5 para modelos GPT (incluyendo todas las evaluaciones); ~\$40 para Claude 3.5 Sonnet
Evidencia de robustez	ninguna defensa integrada en AgentDojo previene completamente la fuga; las defensas pueden reducir ASR a cero en algunos casos; la fuga de contraseñas es menos probable que la de otros datos personales
Modos de fallo	los agentes filtran datos personales no críticos incluso cuando evitan contraseñas; las tareas de extracción de datos y autorización son más vulnerables; la probabilidad de fuga de contraseñas aumenta cuando se solicitan detalles adicionales
Código o artefactos	no reportado explícitamente; el estudio usa AgentDojo (público) y un conjunto sintético propio
Conclusión defensiva	Los ataques simples de inyección de instrucciones pueden exfiltrar datos personales observados por agentes LLM durante tareas bancarias simuladas. Aunque los modelos evitan filtrar contraseñas debido a alineaciones de seguridad, siguen siendo vulnerables a la fuga de otros datos personales. Las defensas integradas en AgentDojo reducen pero no eliminan completamente el riesgo. La amenaza es realista y el atacante no necesita adaptabilidad; el impacto en utilidad es significativo (15-50 pp de caída) y el coste de evaluación es moderado.
Confianza de extracción	95

- Hallazgos clave: Los ataques simples de inyección de instrucciones logran una tasa de éxito media del 20% en 16 tareas bancarias simuladas, causando una caída de utilidad del 15-50%. Las defensas integradas en AgentDojo reducen pero no eliminan la fuga de datos personales, aunque los modelos

evitan filtrar contraseñas gracias a alineaciones de seguridad.

- Evidencia de apoyo: In 16 user tasks from AgentDojo, LLMs show a 15-50 percentage point drop in utility under attack, with average attack success rates (ASR) around 20 percent; some defenses reduce ASR to zero.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 10. AgentRedBench: Dynamic Redteaming and Integration-Aware Defense for LLM Agents over SaaS Integrations

AgentRedBench: Dynamic Redteaming and Integration-Aware Defense for LLM Agents over SaaS Integrations entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Arquitectura de agente con llamadas a herramientas (tool-calling loop) sobre integraciones mock. El guardia (AGENTREDGUARD) se interpone entre la respuesta de la integración y el agente. frente a Atacante externo que puede escribir contenido adversarial en una integración (ej. email, wiki, calendario) que el agente lee, buscando que el agente ejecute acciones no autorizadas en otra integración (ataque de inyección indirecta de prompts con escalado de privilegios).. con aplicación en Entre la respuesta de la integración y el procesamiento por el agente (antes de que el agente ejecute acciones).. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: AGENTREDGUARD reduce la tasa de éxito de ataques de inyección indirecta entre 75 y 77 puntos porcentuales en tres familias de modelos, con una tasa de falsos positivos del 0.0% en benignos reales, superando a los guardias de código abierto existentes.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante externo que puede escribir contenido adversarial en una integración (ej. email, wiki, calendario) que el agente lee, buscando que el agente ejecute acciones no autorizadas en otra integración (ataque de inyección indirecta de prompts con escalado de privilegios). mediante Arquitectura de agente con llamadas a herramientas (tool-calling loop) sobre integraciones mock. El guardia (AGENTREDGUARD) se interpone entre la respuesta de la integración y el agente., aplicado en Entre la respuesta de la integración y el procesamiento por el agente (antes de que el agente ejecute acciones).; la capacidad del atacante se clasifica como Alta: el atacante es un LLM (modelo atacante) que genera contenido adversarial dinámicamente para cada escenario, adaptándose al contexto de la integración.. El comparador principal es Llama Guard, PromptGuard 2 y ProtectAI, entre otros. La eficacia se reporta como Sin guardia: 32%-81% en panel de 8 modelos. Con AGENTREDGUARD: reducción de 75-77 puntos porcentuales (pp) en Haiku, GPT-5.4-mini, Gemini-3-flash (ASR no reportado explícitamente, pero se infiere ASR < 10% tras la reducción).., el efecto sobre uso legítimo como no reportado y el sobrecoste como It reduces ASR against no-guard and open-source guard baselines (Llama Guard, PromptGuard 2, ProtectAI, WildGuard) at 0.0% real-benign (0.2% syntheticbenign) false-positive rate and aprox.39 ms median CPU latency.; no reportado. La evidencia de robustez es Validación con retenciones cruzadas: dos familias de atacantes independientes retenidas del entrenamiento; se confirma que la ganancia se transfiere más allá del subconjunto de entrenamiento. y el fallo residual recuperable es No se reportan modos de fallo específicos; se menciona que AGENTREDGUARD supera a las líneas base en ambos ejes (ASR y FPR).. Para la pregunta de investigación, esto importa así: AGENTREDGUARD reduce la tasa de éxito de ataques de inyección indirecta entre 75 y 77 puntos porcentuales en tres familias de modelos, con una tasa de falsos positivos del 0.0% en benignos reales, superando a los guardias de código abierto existentes. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Hiskias Dingeto; William Leeney
- Año: 2026
- DOI: 10.48550/arxiv.2606.02240

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	Claude Sonnet (Anthropic), GPT-5.4-mini (OpenAI), Gemini-3-flash (Google) y Haiku (Anthropic), entre otros (n=8)
Harness o defensa	AGENTREDBENCH / AGENTREDGUARD
Arquitectura de control	Arquitectura de agente con llamadas a herramientas (tool-calling loop) sobre integraciones mock. El guardia (AGENTREDGUARD) se interpone entre la respuesta de la integración y el agente.
Punto de aplicación	Entre la respuesta de la integración y el procesamiento por el agente (antes de que el agente ejecute acciones).
Modelo de amenaza	Atacante externo que puede escribir contenido adversarial en una integración (ej. email, wiki, calendario) que el agente lee, buscando que el agente ejecute acciones no autorizadas en otra integración (ataque de inyección indirecta de prompts con escalado de privilegios).
Tipo de ataque	Inyección indirecta de prompts (indirect prompt injection) con cinco tipos de ataque: no especificados en detalle, pero incluyen escenarios de autorización no especificada.
Adaptatividad del atacante	Alta: el atacante es un LLM (modelo atacante) que genera contenido adversarial dinámicamente para cada escenario, adaptándose al contexto de la integración.
Entorno o benchmark	Entorno simulado con integraciones mock (estado de integración simulado). Evaluación offline con un juez LLM que puntúa las trazas.
Baseline o comparador	Llama Guard, PromptGuard 2 y ProtectAI, entre otros
Métricas de seguridad	Attack Success Rate (ASR); False Positive Rate (FPR) en benignos reales y sintéticos; tasa de finalización de tarea útil; tasa de sobre-rechazo; overhead de tokens; overhead de latencia
Tasa de éxito del ataque	Sin guardia: 32%-81% en panel de 8 modelos. Con AGENTREDGUARD: reducción de 75-77 puntos porcentuales (pp) en Haiku, GPT-5.4-mini, Gemini-3-flash (ASR no reportado explícitamente, pero se infiere $ASR < 10\%$ tras la reducción).
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	It reduces ASR against no-guard and open-source guard baselines (Llama Guard, PromptGuard 2, ProtectAI, WildGuard) at 0.0% real-benign (0.2% syntheticbenign) false-positive rate and aprox.39 ms median CPU latency.
Sobrecoste económico o computacional	no reportado

Campo	Valor
Evidencia de robustez	Validación con retenciones cruzadas: dos familias de atacantes independientes retenidas del entrenamiento; se confirma que la ganancia se transfiere más allá del subconjunto de entrenamiento.
Modos de fallo	No se reportan modos de fallo específicos; se menciona que AGENTREDGUARD supera a las líneas base en ambos ejes (ASR y FPR).
Código o artefactos	Código, esquemas de integración y modelo AGENTREDGUARD publicados abiertamente. Los escenarios canónicos se evalúan a través de un canal mediado por mantenedores con versionado inmutable.
Conclusión defensiva	AGENTREDGUARD reduce significativamente la ASR (75-77pp) frente a la ausencia de guardia en tres familias de modelos, manteniendo un FPR del 0.0% en benignos reales, superando a los guardias de código abierto existentes. La defensa es específica para inyección indirecta en integraciones SaaS, con atacante adaptativo (LLM) y validación de transferencia a atacantes no vistos. El impacto en utilidad y coste no se cuantifica en el resumen, pero se menciona como métrica secundaria.
Confianza de extracción	90

- Hallazgos clave: AGENTREDGUARD reduce la tasa de éxito de ataques de inyección indirecta entre 75 y 77 puntos porcentuales en tres familias de modelos, con una tasa de falsos positivos del 0.0% en benignos reales, superando a los guardias de código abierto existentes. La defensa generaliza a atacantes e integraciones no vistos durante el entrenamiento.
- Evidencia de apoyo: AGENTREDGUARD cuts online attack success by 75–77pp across three target model families (Haiku, GPT-5.4-mini, Gemini-3-flash) at 0.0% real-benign false-positive rate (0.2% on a synthetic-benign corpus), outperforming every open-source baseline with non-trivial...
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 11. The Granularity Mismatch in Agent Security: Argument-Level Provenance Solves Enforcement and Isolates the LLM Reasoning Bottleneck

The Granularity Mismatch in Agent Security: Argument-Level Provenance Solves Enforcement and Isolates the LLM Reasoning Bottleneck entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Monitor en tiempo de ejecución que asigna roles semánticos a argumentos de herramientas, rastrea procedencia de valores y verifica contratos de confianza por rol. frente a Inyección indirecta de instrucciones en contenido no fiable (web, email, archivos, salidas de API) que puede influir en argumentos de herramientas con autoridad., con aplicación en Antes de ejecutar la llamada a la herramienta, comprobando cada argumento contra su contrato de confianza basado en procedencia.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: PACT, un monitor de seguridad a nivel de argumento, alcanza un 100% de seguridad en los tres modelos más fuertes en AgentDojo, recuperando un 38.1–46.4% de utilidad, superando a CaMeL en 8–16 puntos porcentuales.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Inyección indirecta de instrucciones en contenido no fiable (web, email, archivos, salidas de API) que puede influir en argumentos de herramientas con autoridad. mediante Monitor en tiempo de ejecución que asigna roles semánticos a argumentos de herramientas, rastrea procedencia de valores y verifica contratos de confianza por rol., aplicado en Antes de ejecutar la llamada a la herramienta, comprobando cada argumento contra su contrato de confianza basado en procedencia.; la capacidad del atacante se clasifica como No adaptativo (ataques estáticos predefinidos en los benchmarks). El comparador principal es CaMeL, monitores planos a nivel de invocación y oracle provenance (como referencia ideal). La eficacia se reporta como 0% en los tres modelos más fuertes con PACT (100% de seguridad); en modelos más débiles no se reporta ASR exacto pero se indica que PACT alcanza 100% de seguridad en los tres modelos más fuertes., el efecto sobre uso legítimo como PACT recupera 38.1–46.4% de utilidad en los tres modelos más fuertes, 8–16 puntos porcentuales por encima de CaMeL al mismo nivel de seguridad. y el sobrecoste como P99 inferior a 272 microsegundos en 100.000 comprobaciones; throughput entre 13,6K y 18,0K comprobaciones/s.; no reportado. La evidencia de robustez es Ablaciones muestran que tanto roles semánticos como procedencia entre pasos son necesarios para el rendimiento; PACT supera a CaMeL en utilidad al mismo nivel de seguridad. y el fallo residual recuperable es El cuello de botella restante es la inferencia de procedencia y la síntesis de contratos; en modelos más débiles la seguridad puede no ser perfecta.. Para la pregunta de investigación, esto importa así: PACT, un monitor de seguridad a nivel de argumento, alcanza un 100% de seguridad en los tres modelos más fuertes en AgentDojo, recuperando un 38.1–46.4% de utilidad, superando a CaMeL en 8–16 puntos porcentuales. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Linfeng Fan; Ziwei Li; Yuan Tian; Yichen Wang; Rongsheng Li; Xiong Wang
- Año: 2026
- DOI: 10.48550/arxiv.2605.11039

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	PACT, CaMeL, GPT-4 y GPT-4o, entre otros (n=5)
Harness o defensa	PACT (Provenance-Aware Capability Contracts)
Arquitectura de control	Monitor en tiempo de ejecución que asigna roles semánticos a argumentos de herramientas, rastrea procedencia de valores y verifica contratos de confianza por rol.
Punto de aplicación	Antes de ejecutar la llamada a la herramienta, comprobando cada argumento contra su contrato de confianza basado en procedencia.
Modelo de amenaza	Inyección indirecta de instrucciones en contenido no fiable (web, email, archivos, salidas de API) que puede influir en argumentos de herramientas con autoridad.
Tipo de ataque	Indirect prompt injection
Adaptatividad del atacante	No adaptativo (ataques estáticos predefinidos en los benchmarks)
Entorno o benchmark	Benchmark AgentDojo y conjuntos de diagnóstico de confianza mixta; evaluación offline con oracle provenance y con modelos reales.
Baseline o comparador	CaMeL, monitores planos a nivel de invocación y oracle provenance (como referencia ideal)

Campo	Valor
Métricas de seguridad	Attack success rate (ASR); utility (tasa de éxito en tareas benignas); tasa de falsos positivos; tasa de falsos negativos
Tasa de éxito del ataque	0% en los tres modelos más fuertes con PACT (100% de seguridad); en modelos más débiles no se reporta ASR exacto pero se indica que PACT alcanza 100% de seguridad en los tres modelos más fuertes.
Falsos positivos	0 falsos positivos en 267 llamadas benignas durante un despliegue de siete días; los 13 ataques observados fueron bloqueados.
Impacto en utilidad	PACT recupera 38.1–46.4% de utilidad en los tres modelos más fuertes, 8–16 puntos porcentuales por encima de CaMeL al mismo nivel de seguridad.
Sobrecoste de latencia	P99 inferior a 272 microsegundos en 100.000 comprobaciones; throughput entre 13,6K y 18,0K comprobaciones/s.
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Ablaciones muestran que tanto roles semánticos como procedencia entre pasos son necesarios para el rendimiento; PACT supera a CaMeL en utilidad al mismo nivel de seguridad.
Modos de fallo	El cuello de botella restante es la inferencia de procedencia y la síntesis de contratos; en modelos más débiles la seguridad puede no ser perfecta.
Código o artefactos	Sí (se menciona que los experimentos son totalmente reproducibles localmente y que se liberan artefactos)
Conclusión defensiva	PACT alcanza un 100% de seguridad en los tres modelos más fuertes de AgentDojo y recupera un 38,1-46,4% de utilidad, 8-16 puntos por encima de CaMeL al mismo nivel de seguridad. En un despliegue de siete días bloquea 13 ataques sin falsos positivos en 267 llamadas benignas y su monitor mantiene P99 inferior a 272 microsegundos. La evidencia sigue limitada por ataques no adaptativos y por la calidad de inferencia de procedencia y síntesis de contratos.
Confianza de extracción	95

- Hallazgos clave: PACT, un monitor de seguridad a nivel de argumento, alcanza un 100% de seguridad en los tres modelos más fuertes en AgentDojo, recuperando un 38.1–46.4% de utilidad, superando a CaMeL en 8–16 puntos porcentuales. Las ablaciones confirman que tanto los roles semánticos como la procedencia entre pasos son componentes necesarios.
- Evidencia de apoyo: Under oracle provenance, PACT achieves 100% utility and 100% security on mixed-trust diagnostic suites, while flat invocation-level monitors incur false positives or false negatives.
- Localización de la evidencia: pp. 1, 8-9
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 12. Trojan Hippo: Weaponizing Agent Memory for Data Exfiltration

Trojan Hippo: Weaponizing Agent Memory for Data Exfiltration entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Arquitectura de agente con memoria persistente: el agente lee correos, almacena en memoria a largo plazo, y responde a consultas del usuario; backends: explicit tool memory, agentic memory, RAG, sliding-window context frente a Atacante planta carga útil latente mediante una única llamada a herramienta no confiable (correo manipulado); la carga útil permanece dormida hasta que el usuario menciona un tema sensible, momento en el que exfiltra datos al atacante; modelo de amenaza más realista que envenenamiento de memoria previo, con aplicación en Memoria a largo plazo del agente (almacenamiento persistente); activación en tiempo de consulta cuando se discuten temas sensibles. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: El ataque Trojan Hippo demuestra que la memoria persistente en agentes LLM es vulnerable a exfiltración de datos con alta tasa de éxito (85-100%) incluso tras 100 sesiones benignas.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante planta carga útil latente mediante una única llamada a herramienta no confiable (correo manipulado); la carga útil permanece dormida hasta que el usuario menciona un tema sensible, momento en el que exfiltra datos al atacante; modelo de amenaza más realista que envenenamiento de memoria previo mediante Arquitectura de agente con memoria persistente: el agente lee correos, almacena en memoria a largo plazo, y responde a consultas del usuario; backends: explicit tool memory, agentic memory, RAG, sliding-window context, aplicado en Memoria a largo plazo del agente (almacenamiento persistente); activación en tiempo de consulta cuando se discuten temas sensibles; la capacidad del atacante se clasifica como Alta: el benchmark OpenEvolve refina ataques continuamente (adaptive red-teaming). El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia se reporta como Instantiated on an email assistant across four memory backends (explicit tool memory, agentic memory, RAG, and sliding-window context), Trojan Hippo achieves up to 85-100% ASR against current frontier models from OpenAI and Google, with planted memories successfully activating even after 100 benign sessions., el efecto sobre uso legítimo como Varía ampliamente según requisitos de tarea; las defensas reducen ASR pero con costos de utilidad que dependen del perfil de uso y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Las defensas reducen ASR a 0-5%; las memorias plantadas se activan incluso tras 100 sesiones benignas sin defensa y el fallo residual recuperable es Las defensas pueden tener alta utilidad en algunos perfiles de tarea; el compromiso seguridad-utilidad es el principal desafío. Para la pregunta de investigación, esto importa así: El ataque Trojan Hippo demuestra que la memoria persistente en agentes LLM es vulnerable a exfiltración de datos con alta tasa de éxito (85-100%) incluso tras 100 sesiones benignas. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Debeshee Das; Julien Piet; Darya Kaviani; Luca Beurer-Kellner; Florian Tramèr; David Wagner
- Año: 2026
- DOI: 10.48550/arxiv.2605.01970

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	Modelos frontera de OpenAI y Google (no se especifican nombres concretos en el digest) y asistentes de correo electrónico con backends de memoria:

Campo	Valor
Harness o defensa	Trojan Hippo evaluation framework (OpenEvolve-based adaptive red-teaming benchmark + capability-aware security/utility analysis)
Arquitectura de control	Arquitectura de agente con memoria persistente: el agente lee correos, almacena en memoria a largo plazo, y responde a consultas del usuario; backends: explicit tool memory, agentic memory, RAG, sliding-window context
Punto de aplicación	Memoria a largo plazo del agente (almacenamiento persistente); activación en tiempo de consulta cuando se discuten temas sensibles
Modelo de amenaza	Atacante planta carga útil latente mediante una única llamada a herramienta no confiable (correo manipulado); la carga útil permanece dormida hasta que el usuario menciona un tema sensible, momento en el que exfiltra datos al atacante; modelo de amenaza más realista que envenenamiento de memoria previo
Tipo de ataque	Persistent memory attack (Trojan Hippo): plantación de carga útil latente en memoria a largo plazo, activación condicional y exfiltración de datos
Adaptatividad del atacante	Alta: el benchmark OpenEvolve refina ataques continuamente (adaptive red-teaming)
Entorno o benchmark	Simulación de asistente de correo electrónico con múltiples sesiones (100 benignas + ataques); evaluación automatizada
Baseline o comparador	Sin defensa, cuatro defensas inspiradas en principios básicos de seguridad (no se nombran explícitamente en el digest) y comparación entre
Métricas de seguridad	Attack Success Rate (ASR); utility cost (no especificado en detalle en el digest); capacidad de activación tras sesiones benignas
Tasa de éxito del ataque	Instantiated on an email assistant across four memory backends (explicit tool memory, agentic memory, RAG, and sliding-window context), Trojan Hippo achieves up to 85-100% ASR against current frontier models from OpenAI and Google, with planted memories successfully activating even after 100 benign sessions.
Falsos positivos	no reportado
Impacto en utilidad	Varía ampliamente según requisitos de tarea; las defensas reducen ASR pero con costos de utilidad que dependen del perfil de uso
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Las defensas reducen ASR a 0-5%; las memorias plantadas se activan incluso tras 100 sesiones benignas sin defensa
Modos de fallo	Las defensas pueden tener alta utilidad en algunos perfiles de tarea; el compromiso seguridad-utilidad es el principal desafío

Campo	Valor
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	El ataque Trojan Hippo logra hasta 85-100% ASR contra modelos frontera sin defensa, con activación persistente incluso tras 100 sesiones benignas. Las defensas inspiradas en principios básicos reducen ASR a 0-5%, pero con costos de utilidad que varían según la tarea. El despliegue efectivo en el mundo real requiere un análisis consciente de capacidades y perfiles de uso, para lo cual se proporciona el marco de evaluación.
Confianza de extracción	90

- Hallazgos clave: El ataque Trojan Hippo demuestra que la memoria persistente en agentes LLM es vulnerable a exfiltración de datos con alta tasa de éxito (85-100%) incluso tras 100 sesiones benignas. Las defensas reducen el ASR a 0-5% pero con costos de utilidad variables, destacando la necesidad de un despliegue consciente del perfil de uso.
- Evidencia de apoyo: Trojan Hippo achieves up to 85-100% ASR against current frontier models from OpenAI and Google, with planted memories successfully activating even after 100 benign sessions.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 13. AgentVisor: Defending LLM Agents Against Prompt Injection via Semantic Virtualization

AgentVisor: Defending LLM Agents Against Prompt Injection via Semantic Virtualization entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Arquitectura de visor semántico con separación de privilegios, inspirada en virtualización de SO; incluye protocolo de auditoría y autocorrección one-shot. frente a Atacante que inyecta prompts maliciosos directa o indirectamente en datos externos no fiables procesados por el agente LLM., con aplicación en Intercepción de llamadas a herramientas (tool calls) mediante un visor semántico de confianza.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: AgentVisor logra una tasa de éxito de ataque del 0.65% con solo un 1.45% de pérdida de utilidad, superando a todas las defensas baseline en la mayoría de los ataques de inyección directa e indirecta.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc sobre benchmark con ataques de inyección directa e indirecta, comparando múltiples defensas y midiendo ASR, utilidad y robustez.. Protege frente a Atacante que inyecta prompts maliciosos directa o indirectamente en datos externos no fiables procesados por el agente LLM. mediante Arquitectura de visor semántico con separación de privilegios, inspirada en virtualización de SO; incluye protocolo de auditoría y autocorrección one-shot., aplicado en Intercepción de llamadas a herramientas (tool calls) mediante un visor semántico de confianza.; la capacidad del atacante se clasifica como No adaptativo (ataques predefinidos, no se menciona adaptación dinámica). El comparador principal es No Defense, Sandwich y Instructional, entre otros. La eficacia se reporta como Extensive experiments show that AgentVisor reduces the attack success rate to 0.65%, achieving this strong defense while incurring only a 1.45% average decrease in utility relative to the No Defense scenario, demonstrating superior performance compared to existing defense methods., el efecto sobre uso legítimo como Disminución media del 1.45% en utilidad respecto al escenario No Defense (UA de 90.00% vs 87.50% en No Attack) y el sobrecoste como no reportado; no reportado. La evidencia de robustez es AgentVisor mantiene ASR cercano a 0% en 6 de 7 ataques indirectos y ASR de 1.55% en ataque directo;

robustez frente a múltiples variantes de inyección. y el fallo residual recuperable es En ataque SYSTEM, ASR de 0.91% (ligeramente superior); en ataque Important, ASR de 0.00%.. Para la pregunta de investigación, esto importa así: AgentVisor logra una tasa de éxito de ataque del 0.65% con solo un 1.45% de pérdida de utilidad, superando a todas las defensas baseline en la mayoría de los ataques de inyección directa e indirecta. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Zonghao Ying; Haozheng Wang; Jiangfan Liu; Quanchen Zou; Aishan Liu; Jian Yang; Yaodong Yang; Xianglong Liu
- Año: 2026
- DOI: 10.48550/arxiv.2604.24118

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	AgentVisor, No Defense, Sandwich y Instructional, entre otros (n=9)
Harness o defensa	AgentVisor
Arquitectura de control	Arquitectura de visor semántico con separación de privilegios, inspirada en virtualización de SO; incluye protocolo de auditoría y autocorrección one-shot.
Punto de aplicación	Intercepción de llamadas a herramientas (tool calls) mediante un visor semántico de confianza.
Modelo de amenaza	Atacante que inyecta prompts maliciosos directa o indirectamente en datos externos no fiables procesados por el agente LLM.
Tipo de ataque	Direct Prompt Injection; Indirect Prompt Injection (7 variantes: Direct, Ignore, Escape, Fakecom, Combined, SYSTEM, Important)
Adaptatividad del atacante	No adaptativo (ataques predefinidos, no se menciona adaptación dinámica)
Entorno o benchmark	Entorno de laboratorio con benchmark de ataques de inyección; se evalúa ASR y utilidad (UA) en escenarios con y sin ataque.
Baseline o comparador	No Defense, Sandwich y Instructional, entre otros
Métricas de seguridad	Attack Success Rate (ASR); Utility Accuracy (UA); Robustness (medida compuesta)
Tasa de éxito del ataque	Extensive experiments show that AgentVisor reduces the attack success rate to 0.65%, achieving this strong defense while incurring only a 1.45% average decrease in utility relative to the No Defense scenario, demonstrating superior performance compared to existing defense methods.
Falsos positivos	no reportado
Impacto en utilidad	Disminución media del 1.45% en utilidad respecto al escenario No Defense (UA de 90.00% vs 87.50% en No Attack)
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado

Campo	Valor
Evidencia de robustez	AgentVisor mantiene ASR cercano a 0% en 6 de 7 ataques indirectos y ASR de 1.55% en ataque directo; robustez frente a múltiples variantes de inyección.
Modos de fallo	En ataque SYSTEM, ASR de 0.91% (ligeramente superior); en ataque Important, ASR de 0.00%.
Código o artefactos	no reportado
Conclusión defensiva	AgentVisor reduce significativamente la tasa de éxito de ataques de inyección directa e indirecta (ASR 0.65%) con una pérdida mínima de utilidad (1.45%), superando a las defensas baseline (Sandwich, Instructional, Reminder, Isolation, Spotlight, DeBERTa, MELON) en la mayoría de los escenarios. La defensa es robusta frente a 7 variantes de ataque, aunque muestra una ligera vulnerabilidad en el ataque SYSTEM. No se reportan costes de latencia ni computacionales.
Confianza de extracción	95

- Hallazgos clave: AgentVisor logra una tasa de éxito de ataque del 0.65% con solo un 1.45% de pérdida de utilidad, superando a todas las defensas baseline en la mayoría de los ataques de inyección directa e indirecta.
- Evidencia de apoyo: AgentVisor reduces the attack success rate to 0.65%, achieving this strong defense while incurring only a 1.45% average decrease in utility relative to the No Defense scenario
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 14. Aligning Provenance with Authorization: A Dual-Graph Defense for LLM Agents

Aligning Provenance with Authorization: A Dual-Graph Defense for LLM Agents entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Doble grafo: grafo de razonamiento inyectado (procedencia de la ejecución) y grafo de autorización (intención del usuario en contexto limpio); comparación estructural mediante alineación de grafos. frente a Atacante que controla fuentes de datos externas (correos, páginas web, archivos) y realiza inyección indirecta de instrucciones para manipular al agente a realizar operaciones no autorizadas (ej. transferencias de fondos)., con aplicación en Antes de ejecutar la llamada a herramienta, tras construir ambos grafos y realizar la comprobación de alineación.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: AuthGraph reduce la tasa de éxito de ataques de inyección indirecta de instrucciones en agentes LLM de aproximadamente 40% a 1-2% en dos benchmarks, manteniendo una utilidad del 51-76%, superando a defensas previas como CaMeL, DRIFT y Progent.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante que controla fuentes de datos externas (correos, páginas web, archivos) y realiza inyección indirecta de instrucciones para manipular al agente a realizar operaciones no autorizadas (ej. transferencias de fondos). mediante Doble grafo: grafo de razonamiento inyectado (procedencia de la ejecución) y grafo de autorización (intención del usuario en contexto limpio); comparación estructural mediante alineación de grafos., aplicado en Antes de ejecutar la llamada a herramienta, tras construir ambos grafos y realizar la

comprobación de alineación.; la capacidad del atacante se clasifica como No se especifica adaptabilidad del atacante; se asume que el atacante puede incrustar instrucciones maliciosas en datos externos que el agente lee.. El comparador principal es CaMeL, DRIFT y Progent, entre otros. La eficacia se reporta como On AgentDojo, AuthGraph reduces the attack success rate from 40% to 1% while maintaining 76% task completion rate on GPT-4o; on AgentDyn, it reduces the attack success rate from 39% to 2% while preserving 51% utility, outperforming state-of-the-art defenses including CaMeL, DRIFT, and Progent., el efecto sobre uso legítimo como AgentDojo: 76% de finalización de tareas; AgentDyn: 51% de utilidad preservada y el sobre coste como 4,61 s adicionales por tarea (1,87x); comparable a Progent (1,61x) y PromptArmor (2,13x), e inferior a CaMeL (9,21x).; no reportado. La evidencia de robustez es Resultados consistentes en AgentDojo y AgentDyn frente a CaMeL, DRIFT y Progent; no se evalúa un atacante que adapte iterativamente el ataque al conocimiento de AuthGraph. y el fallo residual recuperable es Casos donde el usuario delega explícitamente confianza en fuentes de observación (aprox. 40% de tareas en AgentDyn); se excluyen 2 casos de falsos negativos por decisión del usuario de abrir el límite de confianza.. Para la pregunta de investigación, esto importa así: AuthGraph reduce la tasa de éxito de ataques de inyección indirecta de instrucciones en agentes LLM de aproximadamente 40% a 1-2% en dos benchmarks, manteniendo una utilidad del 51-76%, superando a defensas previas como CaMeL, DRIFT y Progent. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Peiran Wang; Ying Li; Yuan Tian
- Año: 2026
- DOI: 10.48550/arxiv.2605.26497

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-4o, CaMeL, DRIFT y Progent, entre otros
Harness o defensa	AuthGraph
Arquitectura de control	Doble grafo: grafo de razonamiento inyectado (procedencia de la ejecución) y grafo de autorización (intención del usuario en contexto limpio); comparación estructural mediante alineación de grafos.
Punto de aplicación	Antes de ejecutar la llamada a herramienta, tras construir ambos grafos y realizar la comprobación de alineación.
Modelo de amenaza	Atacante que controla fuentes de datos externas (correos, páginas web, archivos) y realiza inyección indirecta de instrucciones para manipular al agente a realizar operaciones no autorizadas (ej. transferencias de fondos).
Tipo de ataque	Indirect prompt injection (inyección indirecta de instrucciones)
Adaptatividad del atacante	No se especifica adaptabilidad del atacante; se asume que el atacante puede incrustar instrucciones maliciosas en datos externos que el agente lee.
Entorno o benchmark	Benchmarks AgentDojo y AgentDyn con tareas de agente que requieren leer datos externos y llamar a herramientas; se mide ataque exitoso si el agente realiza la operación no autorizada.
Baseline o comparador	CaMeL, DRIFT y Progent, entre otros
Métricas de seguridad	Attack Success Rate (ASR); Task Completion Rate (TCR); utilidad (utility)

Campo	Valor
Tasa de éxito del ataque	On AgentDojo, AuthGraph reduces the attack success rate from 40% to 1% while maintaining 76% task completion rate on GPT-4o; on AgentDyn, it reduces the attack success rate from 39% to 2% while preserving 51% utility, outperforming state-of-the-art defenses including CaMeL, DRIFT, and Progent.
Falsos positivos	no reportado
Impacto en utilidad	AgentDojo: 76% de finalización de tareas; AgentDyn: 51% de utilidad preservada
Sobrecoste de latencia	4,61 s adicionales por tarea (1,87x); comparable a Progent (1,61x) y PromptArmor (2,13x), e inferior a CaMeL (9,21x).
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Resultados consistentes en AgentDojo y AgentDyn frente a CaMeL, DRIFT y Progent; no se evalúa un atacante que adapte iterativamente el ataque al conocimiento de AuthGraph.
Modos de fallo	Casos donde el usuario delega explícitamente confianza en fuentes de observación (aprox. 40% de tareas en AgentDyn); se excluyen 2 casos de falsos negativos por decisión del usuario de abrir el límite de confianza.
Código o artefactos	no reportado
Conclusión defensiva	AuthGraph reduce el ASR del 40% al 1% en AgentDojo y del 39% al 2% en AgentDyn, con utilidad del 76% y 51%, respectivamente, y supera a CaMeL, DRIFT y Progent en esas comparaciones. Añade 4,61 s por tarea (1,87x). Es una señal comparativa prometedora para autorización basada en procedencia, no una superioridad general: no se prueba un atacante adaptativo y no se localizó una liberación de código o artefactos.
Confianza de extracción	95

- Hallazgos clave: AuthGraph reduce la tasa de éxito de ataques de inyección indirecta de instrucciones en agentes LLM de aproximadamente 40% a 1-2% en dos benchmarks, manteniendo una utilidad del 51-76%, superando a defensas previas como CaMeL, DRIFT y Progent.
- Evidencia de apoyo: On AgentDojo, AuthGraph reduces the attack success rate from 40% to 1% while maintaining 76% task completion rate on GPT-4o; on AgentDyn, it reduces the attack success rate from 39% to 2% while preserving 51% utility, outperforming state-of-the-art defenses...
- Localización de la evidencia: pp. 1, 9
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 15. Are AI-assisted Development Tools Immune to Prompt Injection?

Are AI-assisted Development Tools Immune to Prompt Injection? entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Model Context Protocol (MCP) frente a ataque de inyección de instrucciones mediante envenenamiento de herramientas (tool-poisoning),

con aplicación en cliente MCP (validación estática, visibilidad de parámetros, detección de inyecciones, advertencias al usuario, sandboxing de ejecución, registro de auditoría). Metodológicamente se articula como evaluación experimental comparativa post hoc con ataques adversariales sobre siete clientes MCP reales. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental comparativa post hoc con ataques adversariales sobre siete clientes MCP reales. Protege frente a ataque de inyección de instrucciones mediante envenenamiento de herramientas (tool-poisoning) mediante Model Context Protocol (MCP), aplicado en cliente MCP (validación estática, visibilidad de parámetros, detección de inyecciones, advertencias al usuario, sandboxing de ejecución, registro de auditoría); la capacidad del atacante se clasifica como no reportado. El comparador principal es no reportado. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es no reportado y el fallo residual recuperable es envenenamiento cruzado de herramientas, explotación de parámetros ocultos, invocación no autorizada de herramientas. Para la pregunta de investigación, esto importa así: Existen disparidades significativas en la seguridad de los clientes MCP frente a inyección de instrucciones: Claude Desktop muestra fuertes barreras, mientras que Cursor es altamente susceptible a envenenamiento cruzado de herramientas y explotación de parámetros ocultos. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Charoes Huang; Xin Huang; Amin Milani Fard
- Año: 2026
- DOI: 10.48550/arxiv.2603.21642

Campo	Valor
Tipo de trabajo	Empírico
Diseño	evaluación experimental comparativa post hoc con ataques adversariales sobre siete clientes MCP reales
Modelos o sistemas protegidos	Claude Desktop, Claude Code, Cursor y Cline, entre otros (n=7)
Harness o defensa	no reportado
Arquitectura de control	Model Context Protocol (MCP)
Punto de aplicación	cliente MCP (validación estática, visibilidad de parámetros, detección de inyecciones, advertencias al usuario, sandboxing de ejecución, registro de auditoría)
Modelo de amenaza	ataque de inyección de instrucciones mediante envenenamiento de herramientas (tool-poisoning)
Tipo de ataque	inyección de instrucciones (prompt injection) a través de vectores de envenenamiento de herramientas
Adaptatividad del atacante	no reportado
Entorno o benchmark	entorno controlado de laboratorio con ataques adversariales diseñados por los autores
Baseline o comparador	no reportado
Métricas de seguridad	tasa de éxito del ataque; cobertura de funciones de seguridad (validación estática, visibilidad de parámetros, detección de inyecciones, advertencias al usuario, sandboxing de ejecución, registro de auditoría)
Tasa de éxito del ataque	no reportado

Campo	Valor
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	no reportado
Modos de fallo	envenenamiento cruzado de herramientas, explotación de parámetros ocultos, invocación no autorizada de herramientas
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	Los clientes MCP presentan disparidades significativas en su postura de seguridad frente a inyección de instrucciones por envenenamiento de herramientas; Claude Desktop implementa fuertes barreras, mientras que Cursor muestra alta susceptibilidad a envenenamiento cruzado, explotación de parámetros ocultos e invocación no autorizada. Se proporciona orientación práctica para implementadores.
Confianza de extracción	90

- Hallazgos clave: Existen disparidades significativas en la seguridad de los clientes MCP frente a inyección de instrucciones: Claude Desktop muestra fuertes barreras, mientras que Cursor es altamente susceptible a envenenamiento cruzado de herramientas y explotación de parámetros ocultos.
- Evidencia de apoyo: We present the first empirical analysis of prompt injection with the tool-poisoning vulnerability across seven widely used MCP clients
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 16. AttriGuard: Defeating Indirect Prompt Injection in LLM Agents via Causal Attribution of Tool Invocations

AttriGuard: Defeating Indirect Prompt Injection in LLM Agents via Causal Attribution of Tool Invocations entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Arquitectura de defensa en tiempo de ejecución con atribución causal a nivel de acción; combina reproducción en sombra forzada por el profesor, atenuación de control jerárquico y criterio de supervivencia difuso. frente a Inyección Indirecta de Prompts (IPI): adversario incrusta directivas maliciosas en salidas de herramientas no confiables (páginas web, correos) para secuestrar la ejecución del agente., con aplicación en Antes de ejecutar cada llamada a herramienta propuesta por el agente; verificación contrafactual en tiempo de ejecución.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc con pruebas contrafactuales paralelas sobre benchmarks de agentes; comparación de defensas bajo ataques estáticos y adaptativos.. Protege frente a Inyección Indirecta de Prompts (IPI): adversario incrusta directivas maliciosas en salidas de herramientas no confiables (páginas web, correos) para secuestrar la ejecución del agente. mediante Arquitectura de defensa en tiempo de ejecución con atribución causal a nivel de acción; combina reproducción en sombra forzada por el profesor, atenuación de control jerárquico y criterio de supervivencia difuso., aplicado en Antes de ejecutar cada llamada a herramienta propuesta por el agente;

verificación contrafactual en tiempo de ejecución.; la capacidad del atacante se clasifica como Estática (sin adaptación) y adaptativa (optimización basada en gradiente aproximado con conocimiento parcial de la defensa).. El comparador principal es Sin defensa, Erase-and-Check y PromptProtect, entre otros. La eficacia se reporta como Across four LLMs and two agent benchmarks, AttriGuard achieves 0% ASR under static attacks with negligible utility loss and moderate overhead., el efecto sobre uso legítimo como Pérdida de utilidad insignificante (negligible utility loss). y el sobrecoste como no reportado; Crucially, this performance comes at a moderate token cost of aprox. $2\times$ relative to the undefended baseline.. La evidencia de robustez es AttriGuard mantiene 0% ASR bajo ataques estáticos y sigue siendo resiliente bajo ataques adaptativos basados en optimización, donde otras defensas se degradan significativamente. y el fallo residual recuperable es No se reportan modos de fallo específicos; se menciona robustez frente a estocasticidad del LLM.. Para la pregunta de investigación, esto importa así: AttriGuard logra un 0% de ASR bajo ataques estáticos de inyección indirecta de prompts en agentes LLM, con pérdida de utilidad insignificante y sobrecarga moderada, y se mantiene resiliente frente a ataques adaptativos basados en optimización donde otras defensas fallan. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Yu He; Haozhe Zhu; Yiming Li; Shuo Shao; Hongwei Yao; Zhihao Liu; Zhan Qin
- Año: 2026
- DOI: 10.48550/arxiv.2603.10749

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	AttriGuard, GPT-4, GPT-4o y Claude 3.5 Sonnet, entre otros
Harness o defensa	AttriGuard
Arquitectura de control	Arquitectura de defensa en tiempo de ejecución con atribución causal a nivel de acción; combina reproducción en sombra forzada por el profesor, atenuación de control jerárquico y criterio de supervivencia difuso.
Punto de aplicación	Antes de ejecutar cada llamada a herramienta propuesta por el agente; verificación contrafactual en tiempo de ejecución.
Modelo de amenaza	Inyección Indirecta de Prompts (IPI): adversario incrusta directivas maliciosas en salidas de herramientas no confiables (páginas web, correos) para secuestrar la ejecución del agente.
Tipo de ataque	Inyección indirecta de prompts estática y adaptativa basada en optimización (ataque de caja gris con gradiente aproximado).
Adaptatividad del atacante	Estática (sin adaptación) y adaptativa (optimización basada en gradiente aproximado con conocimiento parcial de la defensa).
Entorno o benchmark	Benchmarks de agentes (AgentBench, GAIA) con inyección de instrucciones maliciosas en contenido externo; se evalúa ASR, utilidad y sobrecarga.
Baseline o comparador	Sin defensa, Erase-and-Check y PromptProtect, entre otros
Métricas de seguridad	Attack Success Rate (ASR); utilidad (tasa de finalización de tarea); sobrecarga de latencia y coste.

Campo	Valor
Tasa de éxito del ataque	Across four LLMs and two agent benchmarks, AttriGuard achieves 0% ASR under static attacks with negligible utility loss and moderate overhead.
Falsos positivos	no reportado
Impacto en utilidad	Pérdida de utilidad insignificante (negligible utility loss).
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	Crucially, this performance comes at a moderate token cost of aprox. $2\times$ relative to the undefended baseline.
Evidencia de robustez	AttriGuard mantiene 0% ASR bajo ataques estáticos y sigue siendo resiliente bajo ataques adaptativos basados en optimización, donde otras defensas se degradan significativamente.
Modos de fallo	No se reportan modos de fallo específicos; se menciona robustez frente a estocasticidad del LLM.
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	AttriGuard propone un nuevo paradigma de atribución causal a nivel de acción para defenderse de IPI. Frente a ataques estáticos logra 0% ASR con pérdida de utilidad insignificante y sobrecarga moderada. Bajo ataques adaptativos optimizados, supera a defensas basadas en discriminación semántica de entrada. La conclusión se basa en la comparación con múltiples defensas en dos benchmarks y cuatro LLMs.
Confianza de extracción	95

- Hallazgos clave: AttriGuard logra un 0% de ASR bajo ataques estáticos de inyección indirecta de prompts en agentes LLM, con pérdida de utilidad insignificante y sobrecarga moderada, y se mantiene resiliente frente a ataques adaptativos basados en optimización donde otras defensas fallan.
- Evidencia de apoyo: Across four LLMs and two agent benchmarks, AttriGuard achieves 0% ASR under static attacks with negligible utility loss and moderate overhead.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 17. ReasAlign: Reasoning Enhanced Safety Alignment against Prompt Injection Attack

ReasAlign: Reasoning Enhanced Safety Alignment against Prompt Injection Attack entra en la revisión porque aporta evidencia trazable en la familia contexto, rag y prompt injection indirecta. Evalúa Arquitectura de razonamiento estructurado con pasos de análisis de consulta, detección de instrucciones conflictivas y preservación de tarea; mecanismo de escalado en tiempo de inferencia con modelo juez preferencia-optimizado frente a ataque indirecto de inyección de instrucciones en sistemas agenticos; instrucciones maliciosas incrustadas en datos externos (por ejemplo, reseñas de productos, contenido de correo electrónico), con aplicación en modelo (a nivel de inferencia, antes de generar respuesta). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: ReasAlign reduce la tasa de éxito de ataques de inyección indirecta al 3.6% en CyberSecEval2, manteniendo una utilidad del 94.6%, superando ampliamente a Meta SecAlign (74.4% ASR, 56.4% utilidad).

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a ataque indirecto de inyección de instrucciones en sistemas agenticos; instrucciones maliciosas incrustadas en datos externos (por ejemplo, reseñas de productos, contenido de correo electrónico) mediante Arquitectura de razonamiento estructurado con pasos de análisis de consulta, detección de instrucciones conflictivas y preservación de tarea; mecanismo de escalado en tiempo de inferencia con modelo juez preferencia-optimizado, aplicado en modelo (a nivel de inferencia, antes de generar respuesta); la capacidad del atacante se clasifica como no adaptativo (ataques estáticos predefinidos en benchmarks). El comparador principal es Meta SecAlign y modelo sin defensa. La eficacia se reporta como In terms of security, ReasAlign reduces the ASR from 43.6% to just 3.6% on CySE, and from 74.4% to only 1.1% on SEP., el efecto sobre uso legítimo como 94.6% utility (ReasAlign en CyberSecEval2); 56.4% utility (Meta SecAlign); comparable al modelo sin defensa y el sobrecoste como no reportado; no reportado. La evidencia de robustez es ReasAlign mantiene baja ASR (3.6%) y alta utilidad (94.6%) en CyberSecEval2, superando a Meta SecAlign (74.4% ASR, 56.4% utility); resultados consistentes en otros benchmarks y el fallo residual recuperable es no reportado explícitamente; posible degradación en escenarios no cubiertos por los benchmarks. Para la pregunta de investigación, esto importa así: ReasAlign reduce la tasa de éxito de ataques de inyección indirecta al 3.6% en CyberSecEval2, manteniendo una utilidad del 94.6%, superando ampliamente a Meta SecAlign (74.4% ASR, 56.4% utilidad). La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Hao Li; Yankai Yang; G. Edward Suh; Ning Zhang; Chaowei Xiao
- Año: 2026
- DOI: 10.48550/arxiv.2601.10173

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	ReasAlign, Meta SecAlign y modelo sin defensa (undefended model) (n=3)
Harness o defensa	ReasAlign
Arquitectura de control	Arquitectura de razonamiento estructurado con pasos de análisis de consulta, detección de instrucciones conflictivas y preservación de tarea; mecanismo de escalado en tiempo de inferencia con modelo juez preferencia-optimizado
Punto de aplicación	modelo (a nivel de inferencia, antes de generar respuesta)
Modelo de amenaza	ataque indirecto de inyección de instrucciones en sistemas agenticos; instrucciones maliciosas incrustadas en datos externos (por ejemplo, reseñas de productos, contenido de correo electrónico)
Tipo de ataque	indirect prompt injection
Adaptatividad del atacante	no adaptativo (ataques estáticos predefinidos en benchmarks)
Entorno o benchmark	evaluación offline sobre benchmarks estándar (CyberSecEval2, SEP)
Baseline o comparador	Meta SecAlign y modelo sin defensa
Métricas de seguridad	Attack Success Rate (ASR); Utility (medida de rendimiento en tarea)
Tasa de éxito del ataque	In terms of security, ReasAlign reduces the ASR from 43.6% to just 3.6% on CySE, and from 74.4% to only 1.1% on SEP.

Campo	Valor
Falsos positivos	no reportado
Impacto en utilidad	94.6% utility (ReasAlign en CyberSecEval2); 56.4% utility (Meta SecAlign); comparable al modelo sin defensa
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	ReasAlign mantiene baja ASR (3.6%) y alta utilidad (94.6%) en CyberSecEval2, superando a Meta SecAlign (74.4% ASR, 56.4% utility); resultados consistentes en otros benchmarks
Modos de fallo	no reportado explícitamente; posible degradación en escenarios no cubiertos por los benchmarks
Código o artefactos	Código y resultados disponibles en https://github.com/leolee99/ReasAlign
Conclusión defensiva	ReasAlign, una defensa a nivel de modelo basada en razonamiento estructurado y escalado en tiempo de inferencia con juez preferencia-optimizado, reduce significativamente la tasa de éxito de ataques de inyección indirecta (ASR 3.6% frente a 74.4% de Meta SecAlign) manteniendo una utilidad alta (94.6% frente a 56.4%) en el benchmark CyberSecEval2, demostrando el mejor equilibrio seguridad-utilidad entre las defensas comparadas.
Confianza de extracción	95

- Hallazgos clave: ReasAlign reduce la tasa de éxito de ataques de inyección indirecta al 3.6% en CyberSecEval2, manteniendo una utilidad del 94.6%, superando ampliamente a Meta SecAlign (74.4% ASR, 56.4% utilidad).
- Evidencia de apoyo: On the representative open-ended CyberSecEval2 benchmark, which includes multiple prompt-injected tasks, ReasAlign achieves 94.6% utility and only 3.6% ASR, far surpassing the state-of-the-art defensive model of Meta SecAlign (56.4% utility and 74.4% ASR).
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre contexto, rag y prompt injection indirecta y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 18. MCP-Guard: A Multi-Stage Defense-in-Depth Framework for Securing Model Context Protocol in Agentic AI

MCP-Guard: A Multi-Stage Defense-in-Depth Framework for Securing Model Context Protocol in Agentic AI entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Arquitectura en tres etapas: Stage 1 (escaneo estático ligero), Stage 2 (detector DNN para ataques semánticos), Stage 3 (modelo E5 fine-tuned + árbitro LLM). frente a Ataques adversariales en el paradigma MCP: inyección de prompts, tool poisoning, shadow hijacking, exfiltración de datos, etc., con aplicación en Antes de que el LLM ejecute la herramienta (pre-ejecución); el árbitro decide si se permite o bloquea la interacción.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: MCP-Guard alcanza un 96.01% de precisión en la detección de prompts adversariales con un pipeline de tres etapas, mejorando el F1 en +23.4 puntos promedio frente a LLM standalone y logrando un speedup de 2.04x en latencia.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo.

Protege frente a Ataques adversariales en el paradigma MCP: inyección de prompts, tool poisoning, shadow hijacking, exfiltración de datos, etc. mediante Arquitectura en tres etapas: Stage 1 (escaneo estático ligero), Stage 2 (detector DNN para ataques semánticos), Stage 3 (modelo E5 fine-tuned + árbitro LLM)., aplicado en Antes de que el LLM ejecute la herramienta (pre-ejecución); el árbitro decide si se permite o bloquea la interacción.; la capacidad del atacante se clasifica como No se especifica adaptividad del atacante; el benchmark incluye ataques estáticos y semánticos generados por GPT-4.. El comparador principal es Modelos LLM standalone (sin pipeline) y MCP-Shield (reportado en literatura). La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como Latencia promedio del pipeline MCP-Guard: 455.9 ms (vs. 930.0 ms en LLM standalone); speedup promedio 2.04x.; no reportado. La evidencia de robustez es El pipeline mejora F1 en +23.4 puntos promedio frente a LLM standalone; recall aumenta de 70.2% a 98.5%. y el fallo residual recuperable es No se reportan modos de fallo específicos; se observa que modelos pequeños (Tynyllama, Llama2) tienen bajo recall standalone que mejora drásticamente con el pipeline.. Para la pregunta de investigación, esto importa así: MCP-Guard alcanza un 96.01% de precisión en la detección de prompts adversariales con un pipeline de tres etapas, mejorando el F1 en +23.4 puntos promedio frente a LLM standalone y logrando un speedup de 2.04x en latencia. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Wenpeng Xing; Zhonghao Qi; Yupeng Qin; Yilin Li; Caini Chang; Jiahui Yu; Changting Lin; Zhenzhen Xie; Meng Han
- Año: 2025
- DOI: 10.48550/arxiv.2508.10991

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-4o-mini, Deepseek-chat, Mistral:7B y Qwen2.5:0.5B, entre otros (n=9)
Harness o defensa	MCP-Guard
Arquitectura de control	Arquitectura en tres etapas: Stage 1 (escaneo estático ligero), Stage 2 (detector DNN para ataques semánticos), Stage 3 (modelo E5 fine-tuned + árbitro LLM).
Punto de aplicación	Antes de que el LLM ejecute la herramienta (pre-ejecución); el árbitro decide si se permite o bloquea la interacción.
Modelo de amenaza	Ataques adversariales en el paradigma MCP: inyección de prompts, tool poisoning, shadow hijacking, exfiltración de datos, etc.
Tipo de ataque	Prompt injection; tool poisoning; shadow hijacking; ataques semánticos; amenazas explícitas (overt threats).
Adaptatividad del atacante	No se especifica adaptividad del atacante; el benchmark incluye ataques estáticos y semánticos generados por GPT-4.
Entorno o benchmark	Benchmark offline con muestras etiquetadas; comparación de precisión, recall, F1, tiempo de inferencia.
Baseline o comparador	Modelos LLM standalone (sin pipeline) y MCP-Shield (reportado en literatura)
Métricas de seguridad	Accuracy; Precision; Recall; F1; Latencia (ms); Speedup.

Campo	Valor
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	Latencia promedio del pipeline MCP-Guard: 455.9 ms (vs. 930.0 ms en LLM standalone); speedup promedio 2.04x.
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	El pipeline mejora F1 en +23.4 puntos promedio frente a LLM standalone; recall aumenta de 70.2% a 98.5%.
Modos de fallo	No se reportan modos de fallo específicos; se observa que modelos pequeños (Tinyllama, Llama2) tienen bajo recall standalone que mejora drásticamente con el pipeline.
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	MCP-Guard reduce significativamente la tasa de éxito de ataques (ASR inferido por alto recall) en el paradigma MCP, con un overhead de latencia moderado (455.9 ms) y speedup frente a LLM standalone. La mejora es robusta en múltiples modelos base, pero no se evalúa el impacto en utilidad ni coste económico. La adaptatividad del atacante no se considera explícitamente.
Confianza de extracción	95

- Hallazgos clave: MCP-Guard alcanza un 96.01% de precisión en la detección de prompts adversariales con un pipeline de tres etapas, mejorando el F1 en +23.4 puntos promedio frente a LLM standalone y logrando un speedup de 2.04x en latencia.
- Evidencia de apoyo: MCP-GUARD employs a three-stage detection pipeline that balances efficiency with accuracy: it progresses from lightweight static scanning for overt threats and a deep neural detector for semantic attacks, to our fine-tuned E5-based model which achieves 96.01%...
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 19. AgenTRIM: Tool Risk Mitigation for Agentic AI

AgenTRIM: Tool Risk Mitigation for Agentic AI entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa guardrail por políticas; control de herramientas; monitor de ejecución frente a riesgos de agencia impulsada por herramientas: agencia excesiva (permisos innecesarios) e insuficiente (falta de invocación de herramientas necesarias); inyección indirecta de instrucciones y mal uso de herramientas, con aplicación en por paso (per-step); filtrado adaptativo y validación consciente del estado de las llamadas a herramientas en tiempo de ejecución. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: AgenTRIM reduce sustancialmente la tasa de éxito de ataques en el benchmark AgentDojo sin degradar el rendimiento de la tarea, mostrando robustez frente a ataques basados en descripciones y capacidad de aplicar políticas de seguridad explícitas.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental post hoc sobre benchmark con fases offline y online; comparación de tasas de éxito de ataque y rendimiento de tarea

frente a líneas base. Protege frente a riesgos de agencia impulsada por herramientas: agencia excesiva (permisos innecesarios) e insuficiente (falta de invocación de herramientas necesarias); inyección indirecta de instrucciones y mal uso de herramientas mediante guardrail por políticas; control de herramientas; monitor de ejecución, aplicado en por paso (per-step); filtrado adaptativo y validación consciente del estado de las llamadas a herramientas en tiempo de ejecución; la capacidad del atacante se clasifica como no reportado. El comparador principal es no reportado explícitamente en el digest y se mencionan defensas existentes (guardrails, filtros, reglas) como comparación cualitativa. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como We evaluate AGEN TRIM on the AgentDojo benchmark (Debenedetti et al., 2024), demonstrating low attack success rate while maintaining high utility, both with and without attacks. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es robusto frente a ataques basados en descripciones; aplicación efectiva de políticas de seguridad explícitas y el fallo residual recuperable es Starting from this entry point, analysis traverses the project directory to enumerate candidate tools, Tb0 = Estatic (C), (1) where Tb0 may contain missing tools, false positives, or incorrect specifications.. Para la pregunta de investigación, esto importa así: AgenTRIM reduce sustancialmente la tasa de éxito de ataques en el benchmark AgentDojo sin degradar el rendimiento de la tarea, mostrando robustez frente a ataques basados en descripciones y capacidad de aplicar políticas de seguridad explícitas. La confianza de extracción es alta y permite integrar el estudio en la comparación central con una base empírica suficiente.

- Autores: Roy Betser; Shamik Bose; Amit Giloni; Chiara Picardi; Sindhu Padakandla; Roman Vainshtein
- Año: 2026
- DOI: 10.48550/arxiv.2601.12449

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	AgenTRIM, GPT-4 (presumiblemente, no se especifica en el digest) y otros modelos no listados explícitamente (n=1)
Harness o defensa	AgenTRIM
Arquitectura de control	guardrail por políticas; control de herramientas; monitor de ejecución
Punto de aplicación	por paso (per-step); filtrado adaptativo y validación consciente del estado de las llamadas a herramientas en tiempo de ejecución
Modelo de amenaza	riesgos de agencia impulsada por herramientas: agencia excesiva (permisos innecesarios) e insuficiente (falta de invocación de herramientas necesarias); inyección indirecta de instrucciones y mal uso de herramientas
Tipo de ataque	indirect prompt injection; tool misuse; description-based attacks
Adaptatividad del atacante	no reportado
Entorno o benchmark	benchmark AgentDojo; experimentos adicionales con ataques basados en descripciones y políticas de seguridad explícitas
Baseline o comparador	no reportado explícitamente en el digest y se mencionan defensas existentes (guardrails, filtros, reglas) como comparación cualitativa
Métricas de seguridad	attack success rate (ASR); task performance (rendimiento de tarea)
Tasa de éxito del ataque	no reportado

Campo	Valor
Falsos positivos	no reportado
Impacto en utilidad	We evaluate AGEN TRIM on the AgentDojo benchmark (Debenedetti et al., 2024), demonstrating low attack success rate while maintaining high utility, both with and without attacks.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	robusto frente a ataques basados en descripciones; aplicación efectiva de políticas de seguridad explícitas
Modos de fallo	Starting from this entry point, analysis traverses the project directory to enumerate candidate tools, Tb0 = Estatic (C), (1) where Tb0 may contain missing tools, false positives, or incorrect specifications.
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	AgenTRIM reduce sustancialmente el éxito de ataques (ASR) en el benchmark AgentDojo sin comprometer el rendimiento de la tarea, demostrando robustez frente a ataques basados en descripciones y capacidad de aplicar políticas de seguridad explícitas. La amenaza abordada es la agencia desequilibrada (excesiva/insuficiente) que amplifica la superficie de ataque. No se reportan líneas base cuantitativas ni adaptatividad del atacante; el efecto en utilidad es mínimo (alto rendimiento mantenido) y no se cuantifican sobrecostos de latencia o coste.
Confianza de extracción	85

- Hallazgos clave: AgenTRIM reduce sustancialmente la tasa de éxito de ataques en el benchmark AgentDojo sin degradar el rendimiento de la tarea, mostrando robustez frente a ataques basados en descripciones y capacidad de aplicar políticas de seguridad explícitas.
- Evidencia de apoyo: Evaluating on the AgentDojo benchmark, AgenTRIM substantially reduces attack success while maintaining high task performance.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 20. Constitutional Classifiers++: Efficient Production-Grade Defenses against Universal Jailbreaks

Constitutional Classifiers++: Efficient Production-Grade Defenses against Universal Jailbreaks entra en la revisión porque aporta evidencia trazable en la familia jailbreak y cumplimiento de políticas. Evalúa Cascada de dos etapas: clasificadores ligeros (probes lineales y clasificadores externos pequeños) filtran tráfico; solo intercambios sospechosos se escalan a clasificadores de intercambio más costosos. frente a atacante que busca jailbreaks universales (prompts maliciosos que evaden defensas para obtener respuestas detalladas a consultas prohibidas), con aplicación en post-procesamiento de respuesta (evaluación de la respuesta generada antes de devolverla al usuario). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: El sistema Constitutional Classifiers++ logra una reducción de 40x en coste

computacional y una tasa de rechazo del 0.05% en producción, mientras que ningún ataque en 1.700 horas de red-teaming logró jailbreaks universales exitosos.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental de defensas mediante red-teaming y comparación de variantes de clasificadores (probes, externos, cascada) en un entorno de producción simulado.. Protege frente a atacante que busca jailbreaks universales (prompts maliciosos que evaden defensas para obtener respuestas detalladas a consultas prohibidas) mediante Cascada de dos etapas: clasificadores ligeros (probes lineales y clasificadores externos pequeños) filtran tráfico; solo intercambios sospechosos se escalan a clasificadores de intercambio más costosos., aplicado en post-procesamiento de respuesta (evaluación de la respuesta generada antes de devolverla al usuario); la capacidad del atacante se clasifica como no reportado (red-teaming humano con iteración, pero no se especifica adaptividad automatizada). El comparador principal es clasificador de intercambio base (sin cascada ni probes) y modelo sin defensa. La eficacia se reporta como 6 2 Best Probe Small External Classifier Extra-Small External 10 0 (c) Probe Layers Attack Success Rate (%) 4 0 (b) Loss Function Ablation Attack Success Rate (%) Attack Success Rate (%) (a) Probe vs External Classifiers Smoothed Softmax Only Neither Softmax Only Smoothed Loss Function 2 0 All Every Second Probe Layers, el efecto sobre uso legítimo como tasa de rechazo del 0.05% en tráfico de producción (impacto mínimo en utilidad) y el sobrecoste como no reportado; reducción de 40x en coste computacional respecto al clasificador de intercambio base. La evidencia de robustez es ningún ataque en 1.700 horas de red-teaming logró respuestas detalladas a las 8 consultas objetivo; se mantiene robustez frente a jailbreaks universales y el fallo residual recuperable es no reportado explícitamente; se menciona que clasificadores que examinan salidas de forma aislada son vulnerables (abordado con clasificadores de intercambio). Para la pregunta de investigación, esto importa así: El sistema Constitutional Classifiers++ logra una reducción de 40x en coste computacional y una tasa de rechazo del 0.05% en producción, mientras que ningún ataque en 1.700 horas de red-teaming logró jailbreaks universales exitosos. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Hoagy Cunningham; Jerry Wei; Zihan Wang; Andrew Persic; Alwin Peng; Jordan Abderrachid; Raj Agarwal; Bobby Chen; Austin Cohen; Andy Dau; Alek Dimitriev; Rob Gilson; Logan Howard; Yijin Hua; Jared Kaplan; Jan Leike; Mu Lin; Christopher Liu; Vladimir Mikulik; Rohit Mitapalli; Clare O'Hara; Jin Pan; Nikhil Saxena; Alex Silverstein; Yue Song; Xunjie Yu; Giulio Zhou; Ethan Perez; Mrinank Sharma
- Año: 2026
- DOI: 10.48550/arxiv.2601.04603

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	Constitutional Classifiers (intercambio, cascada, probes lineales, clasificadores externos) y modelo base no especificado (n=7)
Harness o defensa	Constitutional Classifiers++
Arquitectura de control	Cascada de dos etapas: clasificadores ligeros (probes lineales y clasificadores externos pequeños) filtran tráfico; solo intercambios sospechosos se escalan a clasificadores de intercambio más costosos.
Punto de aplicación	post-procesamiento de respuesta (evaluación de la respuesta generada antes de devolverla al usuario)
Modelo de amenaza	atacante que busca jailbreaks universales (prompts maliciosos que evaden defensas para obtener respuestas detalladas a consultas prohibidas)

Campo	Valor
Tipo de ataque	jailbreak universal (prompts diseñados para eludir restricciones de seguridad)
Adaptatividad del atacante	no reportado (red-teaming humano con iteración, pero no se especifica adaptividad automatizada)
Entorno o benchmark	red-teaming humano (más de 1.700 horas) y evaluación automatizada con consultas objetivo
Baseline o comparador	clasificador de intercambio base (sin cascada ni probes) y modelo sin defensa
Métricas de seguridad	Attack Success Rate (ASR); tasa de rechazo (refusal rate); coste computacional (factor de reducción 40x)
Tasa de éxito del ataque	6 2 Best Probe Small External Classifier Extra-Small External 10 0 (c) Probe Layers Attack Success Rate (%) 4 0 (b) Loss Function Ablation Attack Success Rate (%) Attack Success Rate (%) (a) Probe vs External Classifiers Smoothed Softmax Only Neither Softmax Only Smoothed Loss Function 2 0 All Every Second Probe Layers
Falsos positivos	no reportado
Impacto en utilidad	tasa de rechazo del 0.05% en tráfico de producción (impacto mínimo en utilidad)
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	reducción de 40x en coste computacional respecto al clasificador de intercambio base
Evidencia de robustez	ningún ataque en 1.700 horas de red-teaming logró respuestas detalladas a las 8 consultas objetivo; se mantiene robustez frente a jailbreaks universales
Modos de fallo	no reportado explícitamente; se menciona que clasificadores que examinan salidas de forma aislada son vulnerables (abordado con clasificadores de intercambio)
Código o artefactos	no reportado
Conclusión defensiva	Los Clasificadores Constitucionales++ ofrecen defensa robusta contra jailbreaks universales con coste computacional 40x menor y tasa de rechazo del 0.05%, superando a defensas previas que examinan respuestas de forma aislada. La cascada de clasificadores y el uso de probes lineales son clave para la eficiencia.
Confianza de extracción	90

- Hallazgos clave: El sistema Constitutional Classifiers++ logra una reducción de 40x en coste computacional y una tasa de rechazo del 0.05% en producción, mientras que ningún ataque en 1.700 horas de red-teaming logró jailbreaks universales exitosos.
- Evidencia de apoyo: no attack on this system successfully elicited responses to all eight target queries comparable in detail to an undefended model
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre jailbreak y cumplimiento de políticas y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 21. Guaranteed Jailbreaking Defense via Disrupt-and-Rectify Smoothing

Guaranteed Jailbreaking Defense via Disrupt-and-Rectify Smoothing entra en la revisión porque aporta evidencia trazable en la familia jailbreak y cumplimiento de políticas. Evalúa filtro o clasificador frente a ataques de jailbreaking a nivel de token (GCG) y a nivel de prompt (PAIR), incluyendo ataques adaptativos que conocen la defensa, con aplicación en pre-procesamiento del prompt de entrada (antes de la generación del LLM). Metodológicamente se articula como evaluación experimental comparativa con ataques establecidos y adaptativos sobre múltiples LLMs y conjuntos de prompts. Su aportación central es: DR-Smoothing, mediante disrupción y rectificación de prompts, logra una defensa garantizada contra jailbreaking superando a métodos previos en inocuidad y utilidad, incluso frente a ataques adaptativos.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental comparativa con ataques establecidos y adaptativos sobre múltiples LLMs y conjuntos de prompts. Protege frente a ataques de jailbreaking a nivel de token (GCG) y a nivel de prompt (PAIR), incluyendo ataques adaptativos que conocen la defensa mediante filtro o clasificador, aplicado en pre-procesamiento del prompt de entrada (antes de la generación del LLM); la capacidad del atacante se clasifica como adaptativo: se evalúa con ataques que conocen la defensa (adaptive GCG y adaptive PAIR). El comparador principal es SmoothLLM, Self-Reminder y ICD, entre otros. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como se reporta que DR-Smoothing mantiene alta utilidad (helpfulness) en comparación con baselines; se proporcionan métricas de utilidad (ej. tasa de respuesta útil) y el sobrecoste como no reportado; no reportado. La evidencia de robustez es se demuestra robustez frente a ataques adaptativos (adaptive GCG y adaptive PAIR) y se proporciona un análisis teórico con cotas ajustadas para la probabilidad de éxito de la defensa y el fallo residual recuperable es se identifican casos de fallo en la defensa; se analiza la distancia de movimiento en el espacio de embeddings entre casos exitosos y fallidos. Para la pregunta de investigación, esto importa así: DR-Smoothing, mediante disrupción y rectificación de prompts, logra una defensa garantizada contra jailbreaking superando a métodos previos en inocuidad y utilidad, incluso frente a ataques adaptativos. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Zheng Lin; Zhenxing Niu; Haoxuan Ji; Haichang Gao
- Año: 2026
- DOI: 10.48550/arxiv.2605.10582

Campo	Valor
Tipo de trabajo	Empírico
Diseño	evaluación experimental comparativa con ataques establecidos y adaptativos sobre múltiples LLMs y conjuntos de prompts
Modelos o sistemas protegidos	LLaMA-2-7B-Chat, Vicuna-7B-v1.5, GPT-3.5-turbo y GPT-4
Harness o defensa	DR-Smoothing (Disrupt-and-Rectify Smoothing)
Arquitectura de control	filtro o clasificador
Punto de aplicación	pre-procesamiento del prompt de entrada (antes de la generación del LLM)
Modelo de amenaza	ataques de jailbreaking a nivel de token (GCG) y a nivel de prompt (PAIR), incluyendo ataques adaptativos que conocen la defensa
Tipo de ataque	jailbreaking: GCG (token-level) y PAIR (prompt-level)
Adaptatividad del atacante	adaptativo: se evalúa con ataques que conocen la defensa (adaptive GCG y adaptive PAIR)

Campo	Valor
Entorno o benchmark	experimentos sobre LLMs con prompts jailbreak de benchmarks establecidos
Baseline o comparador	SmoothLLM, Self-Reminder y ICD, entre otros
Métricas de seguridad	tasa de éxito de defensa (defense success rate); tasa de rechazo de respuestas dañinas (harmlessness); utilidad medida como tasa de respuesta útil (helpfulness)
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	se reporta que DR-Smoothing mantiene alta utilidad (helpfulness) en comparación con baselines; se proporcionan métricas de utilidad (ej. tasa de respuesta útil)
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	se demuestra robustez frente a ataques adaptativos (adaptive GCG y adaptive PAIR) y se proporciona un análisis teórico con cotas ajustadas para la probabilidad de éxito de la defensa
Modos de fallo	se identifican casos de fallo en la defensa; se analiza la distancia de movimiento en el espacio de embeddings entre casos exitosos y fallidos
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	DR-Smoothing supera a las defensas baseline (SmoothLLM, Self-Reminder, ICD, Self-Examination) tanto en inocuidad como en utilidad frente a ataques de jailbreaking establecidos y adaptativos, respaldado por un análisis teórico que proporciona cotas ajustadas para la probabilidad de éxito de la defensa.
Confianza de extracción	90

- Hallazgos clave: DR-Smoothing, mediante disrupción y rectificación de prompts, logra una defensa garantizada contra jailbreaking superando a métodos previos en inocuidad y utilidad, incluso frente a ataques adaptativos.
- Evidencia de apoyo: Our approach can defend against both token-level and prompt-level jailbreaking attacks, under both established and adaptive attacking scenarios. Extensive experiments demonstrate that our approach surpasses current state-of-the-art defense methods in terms of...
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre jailbreak y cumplimiento de políticas y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 22. MEMSAD: Gradient-Coupled Anomaly Detection for Memory Poisoning in Retrieval-Augmented Agents

MEMSAD: Gradient-Coupled Anomaly Detection for Memory Poisoning in Retrieval-Augmented Agents entra en la revisión porque aporta evidencia trazable en la familia contexto, rag y prompt injection indirecta. Evalúa Agente aumentado por recuperación con memoria externa persistente (Mem0, FAISS) frente a Envenenamiento de memoria persistente en agentes LLM; modelo de adversario con acceso creciente (Stackelberg game): tres clases de ataque con supuestos de acceso escalonados, con aplicación en Punto de recuperación

de memoria (antes de la generación de respuesta). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: MEMSAD logra TPR=1.00 y FPR=0.00 frente a ataques continuos de envenenamiento de memoria, pero la sustitución de sinónimos discretos evade la detección sin afectar al ASR-R, revelando una limitación fundamental de las defensas en espacio continuo.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Envenenamiento de memoria persistente en agentes LLM; modelo de adversario con acceso creciente (Stackelberg game): tres clases de ataque con supuestos de acceso escalonados mediante Agente aumentado por recuperación con memoria externa persistente (Mem0, FAISS), aplicado en Punto de recuperación de memoria (antes de la generación de respuesta); la capacidad del atacante se clasifica como No adaptativo (ataques fijos precomputados); el adversario no se adapta durante la evaluación. El comparador principal es FAISS (almacén de vectores sin procesar), Mem0 (con reformulación LLM) y clasificador JSON zero-shot con GPT-4o-mini, entre otros. La eficacia se reporta como ASR-R: 0.25 a 1.00 tras corrección de protocolo; para defensas compuestas TPR=1.00, FPR=0.00; synonym substitution DeltaASR-R aprox. 0, el efecto sobre uso legítimo como no reportado y el sobrecoste como (iii) Latency: write-time filtering pays aprox.2 ms per ingestion event but removes per-query overhead at retrieval; post-retrieval consensus pays a recurring per-query cost that scales with k (top-k retrieved entries).; no reportado. La evidencia de robustez es Certificación teórica de radio de detección; optimalidad minimax vía Le Cam; cotas de arrepentimiento en línea; experimentos con 20 ensayos y n=1000 confirman TPR=1.00, FPR=0.00 en todos los ataques continuos y el fallo residual recuperable es Synonym substitution (sustitución de sinónimos discretos) evade detección con DeltaASR-R aprox. 0; laguna de invariancia de sinónimos que las defensas basadas en embeddings no pueden cerrar. Para la pregunta de investigación, esto importa así: MEMSAD logra TPR=1.00 y FPR=0.00 frente a ataques continuos de envenenamiento de memoria, pero la sustitución de sinónimos discretos evade la detección sin afectar al ASR-R, revelando una limitación fundamental de las defensas en espacio continuo. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Ishrith Gowda
- Año: 2026
- DOI: 10.48550/arxiv.2605.03482

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	MEMSAD, GPT-4o-mini, GPT-2 y Mem0, entre otros (n=5)
Harness o defensa	MEMSAD (Semantic Anomaly Detection)
Arquitectura de control	Agente aumentado por recuperación con memoria externa persistente (Mem0, FAISS)
Punto de aplicación	Punto de recuperación de memoria (antes de la generación de respuesta)
Modelo de amenaza	Envenenamiento de memoria persistente en agentes LLM; modelo de adversario con acceso creciente (Stackelberg game): tres clases de ataque con supuestos de acceso escalonados
Tipo de ataque	AgentPoison; Minja; InjecMem; synonym substitution (ataque de evasión discreta)
Adaptatividad del atacante	No adaptativo (ataques fijos precomputados); el adversario no se adapta durante la evaluación
Entorno o benchmark	Entorno simulado con corpus de memoria de tamaño variable (200 y 1000 entradas); evaluación offline con consultas predefinidas

Campo	Valor
Baseline o comparador	FAISS (almacén de vectores sin procesar), Mem0 (con reformulación LLM) y clasificador JSON zero-shot con GPT-4o-mini, entre otros
Métricas de seguridad	ASR-R (Attack Success Rate on Retrieval); TPR (True Positive Rate); FPR (False Positive Rate); AUROC; latencia por entrada
Tasa de éxito del ataque	ASR-R: 0.25 a 1.00 tras corrección de protocolo; para defensas compuestas TPR=1.00, FPR=0.00; synonym substitution DeltaASR-R aprox. 0
Falsos positivos	0.00 para defensas compuestas; 0.00 para clasificador JSON zero-shot (TPR=1.00, FPR=0.00 en AgentPoison, Minja, InjecMem)
Impacto en utilidad	no reportado
Sobrecoste de latencia	(iii) Latency: write-time filtering pays aprox.2 ms per ingestion event but removes per-query overhead at retrieval; post-retrieval consensus pays a recurring per-query cost that scales with k (top-k retrieved entries).
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Certificación teórica de radio de detección; optimalidad minimax vía Le Cam; cotas de arrepentimiento en línea; experimentos con 20 ensayos y n=1000 confirman TPR=1.00, FPR=0.00 en todos los ataques continuos
Modos de fallo	Synonym substitution (sustitución de sinónimos discretos) evade detección con DeltaASR-R aprox. 0; laguna de invariancia de sinónimos que las defensas basadas en embeddings no pueden cerrar
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	MEMSAD proporciona una defensa certificada contra envenenamiento de memoria continua en agentes aumentados por recuperación, logrando TPR=1.00 y FPR=0.00 frente a ataques continuos (AgentPoison, Minja, InjecMem) con garantías teóricas de acoplamiento de gradiente y optimalidad minimax. Sin embargo, la sustitución de sinónimos discretos evade la detección sin degradar el ASR-R, lo que constituye una limitación fundamental de las defensas en espacio continuo. La evaluación corrige una inconsistencia en el protocolo de Chen et al (2024) que subestimaba el ASR-R en un factor de 4. No se reporta impacto en utilidad, latencia ni coste, por lo que la conclusión de seguridad es positiva para ataques continuos pero incompleta frente a ataques discretos y sin datos de sobrecoste.
Confianza de extracción	95

- Hallazgos clave: MEMSAD logra TPR=1.00 y FPR=0.00 frente a ataques continuos de envenenamiento de memoria, pero la sustitución de sinónimos discretos evade la detección sin afectar al ASR-R, revelando una limitación fundamental de las defensas en espacio continuo.
- Evidencia de apoyo: Experiments on a 3×5 attack-defense matrix with bootstrap confidence intervals, Bonferroni-corrected hypothesis tests, and Clopper-Pearson validation (20 trials, n = 1,000) confirm

- the theory: composite defenses achieve $TPR = 1.00$, $FPR = 0.00$ across all...
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre contexto, rag y prompt injection indirecta y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 23. PARSE: Provenance-Aware Retrieval Sanitization for Professional Domain LLM Agents

PARSE: Provenance-Aware Retrieval Sanitization for Professional Domain LLM Agents entra en la revisión porque aporta evidencia trazable en la familia contexto, rag y prompt injection indirecta. Evalúa Pipeline de saneamiento con clasificación de oraciones por probabilidad de inyección, extracción de hechos estructurados, reescritura y bucle de verificación de consistencia; puerta de directividad que enruta documentos de bajo riesgo a una ruta ligera. frente a Inyección de instrucciones en documentos recuperados de fuentes no confiables (dominio camuflado, imitando vocabulario profesional legítimo), con aplicación en post-recuperación, antes de la presentación al LLM. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Inyección de instrucciones en documentos recuperados de fuentes no confiables (dominio camuflado, imitando vocabulario profesional legítimo) mediante Pipeline de saneamiento con clasificación de oraciones por probabilidad de inyección, extracción de hechos estructurados, reescritura y bucle de verificación de consistencia; puerta de directividad que enruta documentos de bajo riesgo a una ruta ligera., aplicado en post-recuperación, antes de la presentación al LLM; la capacidad del atacante se clasifica como no adaptativo (ataques estáticos predefinidos en el benchmark). El comparador principal es paráfrasis (defensa baseline) y sin defensa (tasa de éxito de ataque 25.4%). La eficacia se reporta como Paraphrasing, the strongest defense on synthetic benchmarks, shows no statistically significant attack success rate reduction on real documents ($p=0.500$) while degrading utility from 91.8% to 82.8%., el efecto sobre uso legítimo como PARSE: 86.9% de utilidad (frente a 91.8% baseline sin defensa); paráfrasis: 82.8% de utilidad y el sobrecoste como no reportado; no reportado cuantitativamente; se menciona que la puerta de directividad concentra el coste computacional en documentos de alto riesgo (59% en ruta ligera). La evidencia de robustez es PARSE logra ASR 15.6% (reducción del 38% vs baseline 25.4%) con significación estadística ($p=0.014$) y potencia adecuada; paráfrasis no muestra reducción significativa ($p=0.500$) y el fallo residual recuperable es no reportado explícitamente; se infiere que documentos de alto riesgo pueden requerir más procesamiento. Para la pregunta de investigación, esto importa así: el estudio aporta evidencia sustantiva que ayuda a delimitar mecanismo, contexto y alcance inferencial. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Aaditya Pai
- Año: 2026
- DOI: 10.48550/arxiv.2606.17467

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	PARSE, paráfrasis (defensa baseline) y Llama Guard 3 (mencionado como clasificador de producción) (n=5)
Harness o defensa	PARSE (Provenance-Aware Retrieval Sanitization)

Campo	Valor
Arquitectura de control	Pipeline de saneamiento con clasificación de oraciones por probabilidad de inyección, extracción de hechos estructurados, reescritura y bucle de verificación de consistencia; puerta de directividad que enruta documentos de bajo riesgo a una ruta ligera.
Punto de aplicación	post-recuperación, antes de la presentación al LLM
Modelo de amenaza	Inyección de instrucciones en documentos recuperados de fuentes no confiables (dominio camuflado, imitando vocabulario profesional legítimo)
Tipo de ataque	inyección de instrucciones (domain-camouflaged injection)
Adaptatividad del atacante	no adaptativo (ataques estáticos predefinidos en el benchmark)
Entorno o benchmark	benchmark de documentos reales con 122 tareas en cinco dominios profesionales
Baseline o comparador	paráfrasis (defensa baseline) y sin defensa (tasa de éxito de ataque 25.4%)
Métricas de seguridad	Attack Success Rate (ASR); utilidad (porcentaje de respuestas correctas); significación estadística (p-valor); potencia estadística
Tasa de éxito del ataque	Paraphrasing, the strongest defense on synthetic benchmarks, shows no statistically significant attack success rate reduction on real documents (p=0.500) while degrading utility from 91.8% to 82.8%.
Falsos positivos	no reportado
Impacto en utilidad	PARSE: 86.9% de utilidad (frente a 91.8% baseline sin defensa); paráfrasis: 82.8% de utilidad
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado cuantitativamente; se menciona que la puerta de directividad concentra el coste computacional en documentos de alto riesgo (59% en ruta ligera)
Evidencia de robustez	PARSE logra ASR 15.6% (reducción del 38% vs baseline 25.4%) con significación estadística (p=0.014) y potencia adecuada; paráfrasis no muestra reducción significativa (p=0.500)
Modos de fallo	no reportado explícitamente; se infiere que documentos de alto riesgo pueden requerir más procesamiento
Código o artefactos	no reportado
Conclusión defensiva	PARSE reduce significativamente la tasa de éxito de ataques de inyección de instrucciones en documentos reales (ASR 15.6% vs 25.4% baseline) manteniendo una utilidad cercana a la línea base (86.9% vs 91.8%), a diferencia de la paráfrasis que no muestra mejora significativa y degrada la utilidad. La defensa debe evaluarse en documentos reales del dominio, no en proxies sintéticos.
Confianza de extracción	95

- Hallazgos clave: PARSE reduce la tasa de éxito de ataques de inyección de instrucciones en documentos reales un 38% respecto a la línea base (ASR 15.6% vs 25.4%) con significación estadística ($p=0.014$) y mantiene una utilidad del 86.9%, superando a la paráfrasis que no muestra mejora significativa y reduce la utilidad al 82.8%. Las defensas deben evaluarse en documentos reales del dominio, no en benchmarks sintéticos.
- Evidencia de apoyo: PARSE achieves 15.6% attack success rate—a 38% reduction versus the 25.4% baseline—at 86.9% utility, the only condition that is both statistically significant ($p=0.014$, adequately powered) and maintains near-baseline utility.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre contexto, rag y prompt injection indirecta y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 24. Revisiting JBSHield: Breaking and Rebuilding Representation-Level Jailbreak Defenses

Revisiting JBSHield: Breaking and Rebuilding Representation-Level Jailbreak Defenses entra en la revisión porque aporta evidencia trazable en la familia jailbreak y cumplimiento de políticas. Evalúa Llama-3-8B con capas de atención causal; JBSHield usa dos clasificadores de concepto (tóxico y jailbreak) sobre representaciones de una capa; RTV usa detección de outliers de Mahalanobis sobre huellas dactilares de múltiples capas frente a adaptativo: el atacante conoce la defensa y optimiza el sufijo adversarial para evadirla; incluye escenario de caja blanca total contra RTV, con aplicación en representaciones ocultas (hidden states) antes de la generación; detección a nivel de prompt. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: JBSHield es vulnerable a ataques adaptativos (JB-GCG) que logran hasta 53.4% ASR, mientras que la nueva defensa RTV alcanza AUROC 0.99 y resiste ataques adaptativos con solo 7% ASR a un costo 13 veces mayor.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental de ataques y defensas adaptativos sobre representaciones ocultas de LLM, con comparación de tasas de éxito y curvas AUROC. Protege frente a adaptativo: el atacante conoce la defensa y optimiza el sufijo adversarial para evadirla; incluye escenario de caja blanca total contra RTV mediante Llama-3-8B con capas de atención causal; JBSHield usa dos clasificadores de concepto (tóxico y jailbreak) sobre representaciones de una capa; RTV usa detección de outliers de Mahalanobis sobre huellas dactilares de múltiples capas, aplicado en representaciones ocultas (hidden states) antes de la generación; detección a nivel de prompt; la capacidad del atacante se clasifica como alta: el atacante adapta el objetivo de optimización para suprimir la dirección de rechazo y regularizar el concepto tóxico; se evalúa también recalibración de la defensa. El comparador principal es JBSHield original (ASR 0% reportado), JBSHield-M y RTV, entre otros. La eficacia se reporta como JBSHield, a recent jailbreak defense with a 0% attack success rate in some settings, detects malicious prompts via two concept signals, a toxic concept and a jailbreak concept., el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es RTV alcanza AUROC 0.99 contra JB-GCG; el mejor ataque adaptativo logra solo 7% ASR; la vulnerabilidad de JBSHield es estructural (persiste tras recalibraciones) y el fallo residual recuperable es JBSHield es vulnerable a ataques adaptativos que optimizan conjuntamente la supresión de dirección de rechazo y la regularización del concepto tóxico; la detección en una sola capa es insuficiente. Para la pregunta de investigación, esto importa así: JBSHield es vulnerable a ataques adaptativos (JB-GCG) que logran hasta 53.4% ASR, mientras que la nueva defensa RTV alcanza AUROC 0.99 y resiste ataques adaptativos con solo 7% ASR a un costo 13 veces mayor. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Kemal Derya; Berk Sunar
- Año: 2026
- DOI: 10.48550/arxiv.2605.03095

Campo	Valor
Tipo de trabajo	Empírico
Diseño	evaluación experimental de ataques y defensas adaptativos sobre representaciones ocultas de LLM, con comparación de tasas de éxito y curvas AUROC
Modelos o sistemas protegidos	Llama-3-8B, JBSshield, JBSshield-M y JB-GCG, entre otros (n=1)
Harness o defensa	JBSshield; Representation Trajectory Verification (RTV)
Arquitectura de control	Llama-3-8B con capas de atención causal; JBSshield usa dos clasificadores de concepto (tóxico y jailbreak) sobre representaciones de una capa; RTV usa detección de outliers de Mahalanobis sobre huellas dactilares de múltiples capas
Punto de aplicación	representaciones ocultas (hidden states) antes de la generación; detección a nivel de prompt
Modelo de amenaza	adaptativo: el atacante conoce la defensa y optimiza el sufijo adversarial para evadirla; incluye escenario de caja blanca total contra RTV
Tipo de ataque	sufijo adversarial optimizado mediante gradientes (JB-GCG, modificación de GCG); ataque adaptativo con conocimiento completo de la defensa
Adaptatividad del atacante	alta: el atacante adapta el objetivo de optimización para suprimir la dirección de rechazo y regularizar el concepto tóxico; se evalúa también recalibración de la defensa
Entorno o benchmark	experimentos controlados sobre Llama-3-8B con cinco configuraciones de ataque; se varían capas y posiciones de tokens para RTV
Baseline o comparador	JBSshield original (ASR 0% reportado), JBSshield-M y RTV, entre otros
Métricas de seguridad	Attack Success Rate (ASR); AUROC; tasa de detección; costo computacional relativo
Tasa de éxito del ataque	JBSshield, a recent jailbreak defense with a 0% attack success rate in some settings, detects malicious prompts via two concept signals, a toxic concept and a jailbreak concept.
Falsos positivos	no reportado explícitamente; se reporta AUROC (0.99 para RTV) que integra FPR
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	RTV alcanza AUROC 0.99 contra JB-GCG; el mejor ataque adaptativo logra solo 7% ASR; la vulnerabilidad de JBSshield es estructural (persiste tras recalibraciones)
Modos de fallo	JBSshield es vulnerable a ataques adaptativos que optimizan conjuntamente la supresión de dirección de rechazo y la regularización del concepto tóxico; la detección en una sola capa es insuficiente
Código o artefactos	disponible tras revisión (según el texto: ‘The code will be available after the reviews’)

Campo	Valor
Conclusión defensiva	JBSHield, a pesar de reportar 0% ASR en entornos no adaptativos, es vulnerable a ataques adaptativos (JB-GCG) que logran hasta 53.4% ASR. La defensa RTV, basada en detección de outliers de Mahalanobis sobre huellas dactilares de múltiples capas, ofrece robustez significativamente mayor (AUROC 0.99) y resiste ataques adaptativos con solo 7% ASR a un costo 13 veces mayor. La conclusión es que la detección no adaptativa no implica robustez bajo modelos de amenaza adaptativos, y que la consistencia de representación en múltiples capas es una base más fiable.
Confianza de extracción	95

- Hallazgos clave: JBSHield es vulnerable a ataques adaptativos (JB-GCG) que logran hasta 53.4% ASR, mientras que la nueva defensa RTV alcanza AUROC 0.99 y resiste ataques adaptativos con solo 7% ASR a un costo 13 veces mayor.
- Evidencia de apoyo: JB-GCG achieves an average ASR of 46.2%, reaching up to 53.4% in the strongest setting.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre jailbreak y cumplimiento de políticas y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 25. SingGuard-NSFA: Extensible Guardrails for Agentic AI via Generative Reasoning and Real-Time Classification

SingGuard-NSFA: Extensible Guardrails for Agentic AI via Generative Reasoning and Real-Time Classification entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Modelo backbone congelado con cabezales de clasificación discriminativos (modo Cls.) y modo generativo SFT (modo Gen.) frente a Amenazas operativas: inyección de prompts, extracción de información sensible, solicitudes de código malicioso, mal uso peligroso de herramientas, agotamiento de recursos (taxonomía NSFA con 185 variantes), con aplicación en Antes de la ejecución de la acción del agente (detección en consultas y respuestas). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: SingGuard-NSFA alcanza $F1 \geq 94\%$ en benchmarks diseñados a propósito, superando en 6-12 puntos a las barreras competidoras.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Amenazas operativas: inyección de prompts, extracción de información sensible, solicitudes de código malicioso, mal uso peligroso de herramientas, agotamiento de recursos (taxonomía NSFA con 185 variantes) mediante Modelo backbone congelado con cabezales de clasificación discriminativos (modo Cls.) y modo generativo SFT (modo Gen.), aplicado en Antes de la ejecución de la acción del agente (detección en consultas y respuestas); la capacidad del atacante se clasifica como No se especifica adaptabilidad del atacante; los ataques son estáticos en los benchmarks. El comparador principal es AgentDoG1.5-FG (4B), Granite Guardian 3.2 (5B) y Granite Guardian 4.1 (8B), entre otros. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como Aproximadamente 50 ms en modo clasificación; no reportado. La evidencia de robustez es Evaluación en 133 idiomas y en 5 datasets públicos; ablaciones muestran que los cabezales de clasificación mantienen rendimiento en dominios no vistos y el fallo residual recuperable es El dominio Resource Abuse en CrossSource-Query tiene solo 2 muestras, por lo que su F1 no es fiable; modo generativo puede fallar en detección de ciertos riesgos (ej. Resource

en modo Gen. obtiene 0% F1 en algunos modelos). Para la pregunta de investigación, esto importa así: SingGuard-NSFA alcanza F1 $\geq 94\%$ en benchmarks diseñados a propósito, superando en 6-12 puntos a las barreras competidoras. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: SingGuard Team
- Año: 2026
- DOI: 10.48550/arxiv.2607.13081

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	SingGuard-NSFA (0.8B, 2B, 4B, 9B), AgentDoG1.5-FG (4B), Granite Guardian 3.2 (5B) y Granite Guardian 4.1 (8B), entre otros (n=11)
Harness o defensa	SingGuard-NSFA (nsfaguard)
Arquitectura de control	Modelo backbone congelado con cabezales de clasificación discriminativos (modo Cls.) y modo generativo SFT (modo Gen.)
Punto de aplicación	Antes de la ejecución de la acción del agente (detección en consultas y respuestas)
Modelo de amenaza	Amenazas operativas: inyección de prompts, extracción de información sensible, solicitudes de código malicioso, mal uso peligroso de herramientas, agotamiento de recursos (taxonomía NSFA con 185 variantes)
Tipo de ataque	Prompt injection; sensitive information extraction; malicious code requests; dangerous tool misuse; resource exhaustion
Adaptatividad del atacante	No se especifica adaptabilidad del atacante; los ataques son estáticos en los benchmarks
Entorno o benchmark	Benchmark offline (purpose-built y cross-source) con muestras de consultas y respuestas
Baseline o comparador	AgentDoG1.5-FG (4B), Granite Guardian 3.2 (5B) y Granite Guardian 4.1 (8B), entre otros
Métricas de seguridad	F1; precisión; recall; F1 por dominio (Danger Ops, Mal. Code, Prompt Inj., Resource, Sensitive Info)
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	Aproximadamente 50 ms en modo clasificación
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Evaluación en 133 idiomas y en 5 datasets públicos; ablaciones muestran que los cabezales de clasificación mantienen rendimiento en dominios no vistos
Modos de fallo	El dominio Resource Abuse en CrossSource-Query tiene solo 2 muestras, por lo que su F1 no es fiable; modo generativo puede fallar en detección de ciertos riesgos (ej. Resource en modo Gen. obtiene 0% F1 en algunos modelos)

Campo	Valor
Código o artefactos	Modelos disponibles en Hugging Face y ModelScope; código en GitHub (https://github.com/inclusionAI/SingGuard-NSFA)
Conclusión defensiva	SingGuard-NSFA supera a las barreras competidoras en F1 (6-12 puntos absolutos) en benchmarks diseñados a propósito, y mantiene rendimiento competitivo en fuentes cruzadas. El modo clasificación ofrece detección en tiempo real con baja latencia. Sin embargo, la evaluación se limita a benchmarks estáticos; no se reporta impacto en utilidad del agente ni coste computacional completo. La adaptabilidad del atacante no se considera.
Confianza de extracción	95

- Hallazgos clave: SingGuard-NSFA alcanza F1 $\geq 94\%$ en benchmarks diseñados a propósito, superando en 6-12 puntos a las barreras competidoras. El modo clasificación permite detección en tiempo real (~50 ms) con rendimiento competitivo en fuentes cruzadas (91.29% F1 para el modelo de 9B).
- Evidencia de apoyo: all achieving $\geq 94\%$ F1 on purpose-built benchmarks and surpassing the strongest competing guardrails by 6 to 12 absolute points
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 26. Think Twice Before You Act: Protecting LLM Agents Against Tool Description Poisoning via Isolated Planning

Think Twice Before You Act: Protecting LLM Agents Against Tool Description Poisoning via Isolated Planning entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Planificación aislada: las invocaciones sospechosas colocan la herramienta en una lista de influencia, rompiendo la influencia cruzada de descripciones envenenadas. frente a Envenenamiento de descripciones entre herramientas (cross-tool description poisoning): el adversario modifica metadatos de herramientas visibles para el planificador, influyendo en la selección de herramientas posteriores sin necesidad de que la herramienta envenenada sea invocada., con aplicación en Durante la planificación, tras detectar invocaciones desalineadas o sospechosas, se aísla la herramienta en la lista de influencia.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Tool-Guard reduce el ASR a valores cercanos a cero (0-2%) frente a envenenamiento de descripciones entre herramientas, manteniendo una alta utilidad de la tarea (por ejemplo, GPT-4o-mini pasa de 64.95% a 68.04% de utilidad).

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc sobre benchmarks con comparación de defensas existentes y propuesta (Tool-Guard) en múltiples modelos y suites de tareas.. Protege frente a Envenenamiento de descripciones entre herramientas (cross-tool description poisoning): el adversario modifica metadatos de herramientas visibles para el planificador, influyendo en la selección de herramientas posteriores sin necesidad de que la herramienta envenenada sea invocada. mediante Planificación aislada: las invocaciones sospechosas colocan la herramienta en una lista de influencia, rompiendo la influencia cruzada de descripciones envenenadas., aplicado en Durante la planificación, tras detectar invocaciones desalineadas o sospechosas, se aísla la herramienta en la lista de influencia.; la capacidad del atacante se clasifica como El adversario puede emplear diferentes estrategias de ataque (Tabla 7) para evitar patrones fijos, y personaliza descripciones envenenadas para engañar a defensas como DRIFT y ProAgent.. El comparador principal es sin defensa, defensas existentes contra inyección de prompts y DRIFT,

entre otros. La eficacia se reporta como ASR sin defensa: 10.31% (GPT-4o-mini), 43.30% (GPT-4o), 31.96% (Gemini-2.5-Pro), 52.58% (Gemini-2.5-Flash), 4.12% (Claude-3.5-Haiku) en promedio ponderado; con Tool-Guard: 1.03%, 2.06%, 1.03%, 0.00%, 1.03% respectivamente., el efecto sobre uso legítimo como La utilidad de la tarea se mantiene alta: por ejemplo, GPT-4o-mini pasa de 64.95% (sin defensa) a 68.04% (con defensa) en promedio; en algunos casos la utilidad incluso mejora ligeramente. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Tool-Guard reduce ASR a valores cercanos a cero en todos los modelos y suites, manteniendo alta utilidad de la tarea. y el fallo residual recuperable es Prior work (e.g., (Invariantlabs, 2025)) mitigates this via static scanning that removes tools with explicit cross-tool references, but influence can arise without explicit mentions (e.g., through high-level task semantics) and many explicit references are benign (e.g., two-step verification), leading to incompleteness. Para la pregunta de investigación, esto importa así: Tool-Guard reduce el ASR a valores cercanos a cero (0-2%) frente a envenenamiento de descripciones entre herramientas, manteniendo una alta utilidad de la tarea (por ejemplo, GPT-4o-mini pasa de 64.95% a 68.04% de utilidad). La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Shanghao Shi; Xiao Wang; Chaoyu Zhang; Hao Li; Wenjing Lou; Thomas Hou; Yevgeniy Vorobeychik; Chongjie Zhang; Ning Zhang
- Año: 2026
- DOI: 10.48550/arxiv.2606.20922

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-4o-mini, GPT-4o, Gemini-2.5-Pro y Gemini-2.5-Flash, entre otros
Harness o defensa	Tool-Guard
Arquitectura de control	Planificación aislada: las invocaciones sospechosas colocan la herramienta en una lista de influencia, rompiendo la influencia cruzada de descripciones envenenadas.
Punto de aplicación	Durante la planificación, tras detectar invocaciones desalineadas o sospechosas, se aísla la herramienta en la lista de influencia.
Modelo de amenaza	Envenenamiento de descripciones entre herramientas (cross-tool description poisoning): el adversario modifica metadatos de herramientas visibles para el planificador, influyendo en la selección de herramientas posteriores sin necesidad de que la herramienta envenenada sea invocada.
Tipo de ataque	Envenenamiento de descripciones de herramientas (tool description poisoning) con instrucciones de autoridad añadidas al final, preservando el contexto original.
Adaptatividad del atacante	El adversario puede emplear diferentes estrategias de ataque (Tabla 7) para evitar patrones fijos, y personaliza descripciones envenenadas para engañar a defensas como DRIFT y ProAgent.
Entorno o benchmark	Benchmarks AgentDojo y ASB con múltiples modelos y suites de tareas (banca, Slack, viajes, espacio de trabajo).
Baseline o comparador	sin defensa, defensas existentes contra inyección de prompts y DRIFT, entre otros

Campo	Valor
Métricas de seguridad	Attack Success Rate (ASR); Task Utility (tasa de finalización de tareas); False Positive Rate (implícito en utilidad)
Tasa de éxito del ataque	ASR sin defensa: 10.31% (GPT-4o-mini), 43.30% (GPT-4o), 31.96% (Gemini-2.5-Pro), 52.58% (Gemini-2.5-Flash), 4.12% (Claude-3.5-Haiku) en promedio ponderado; con Tool-Guard: 1.03%, 2.06%, 1.03%, 0.00%, 1.03% respectivamente.
Falsos positivos	no reportado
Impacto en utilidad	La utilidad de la tarea se mantiene alta: por ejemplo, GPT-4o-mini pasa de 64.95% (sin defensa) a 68.04% (con defensa) en promedio; en algunos casos la utilidad incluso mejora ligeramente.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Tool-Guard reduce ASR a valores cercanos a cero en todos los modelos y suites, manteniendo alta utilidad de la tarea.
Modos de fallo	Prior work (e.g., (Invariantlabs, 2025)) mitigates this via static scanning that removes tools with explicit cross-tool references, but influence can arise without explicit mentions (e.g., through high-level task semantics) and many explicit references are benign (e.g., two-step verification), leading to incompleteness
Código o artefactos	Código disponible en https://github.com/shishishi123/Tool-Guard
Conclusión defensiva	Tool-Guard reduce drásticamente el ASR (de hasta 52.58% a 0-2%) frente a envenenamiento de descripciones entre herramientas, manteniendo la utilidad de la tarea, sin necesidad de sacrificar funcionalidad. La defensa supera a las defensas existentes contra inyección de prompts, que se transfieren mal a esta amenaza.
Confianza de extracción	95

- Hallazgos clave: Tool-Guard reduce el ASR a valores cercanos a cero (0-2%) frente a envenenamiento de descripciones entre herramientas, manteniendo una alta utilidad de la tarea (por ejemplo, GPT-4o-mini pasa de 64.95% a 68.04% de utilidad). Las defensas existentes contra inyección de prompts se transfieren mal a esta amenaza.
- Evidencia de apoyo: Tool-Guard substantially reduces attack success while maintaining high task utility.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 27. Image-embedded prompt injection vulnerability of vision-language models in dental radiology: a cross-vendor attack–defense evaluation

Image-embedded prompt injection vulnerability of vision-language models in dental radiology: a cross-vendor attack–defense evaluation entra en la revisión porque aporta evidencia trazable en la familia monitorización, verificación y salida. Evalúa filtro o clasificador frente a atacante que incrusta texto adversarial en los datos de píxeles de imágenes médicas para alterar la salida del VLM, con aplicación en post-imagen (sanitización de texto en píxeles o escalado a revisión humana). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Los cuatro VLM evaluados son vulnerables a la inyección de prompt incrustada en imagen, con ASR de hasta 62.6% en GPT-4o.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a atacante que incrusta texto adversarial en los datos de píxeles de imágenes médicas para alterar la salida del VLM mediante filtro o clasificador, aplicado en post-imagen (sanitización de texto en píxeles o escalado a revisión humana); la capacidad del atacante se clasifica como no adaptativo (ataques pre-generados, sin adaptación en tiempo real). El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia se reporta como hasta 62.6% (95% CI: 58.5–66.7%) para GPT-4o sin defensa; con OCR sanitization ASR agrupado 0.2%, el efecto sobre uso legítimo como Consequently, the utility-preservation findings reported below (F1 within 0.6 percentage points of baseline) are driven primarily by sensitivity and should not be interpreted as a comprehensive clinical validation of diagnostic robustness across balanced populations. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es OCR-based sanitization logró la mayor reducción de ataque (ASR agrupado 0.2%); ProvDent proporciona mecanismo de gobernanza complementario y el fallo residual recuperable es no reportado explícitamente; se observa que los perfiles de vulnerabilidad son específicos del modelo con interacciones ataque-modelo significativas. Para la pregunta de investigación, esto importa así: Los cuatro VLM evaluados son vulnerables a la inyección de prompt incrustada en imagen, con ASR de hasta 62.6% en GPT-4o. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Babak Saravi; Daman Deep Singh; Lara Schorn; Andreas Vollmer; Christoph Sproll; Norbert Kübler; Felix Schrader
- Año: 2026
- DOI: 10.21203/rs.3.rs-9932271/v1

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-4o, Gemini 2.5 Flash, Claude Sonnet 4.5 y MedGemma 4B
Harness o defensa	ProvDent (provenance-aware defense with bounded clinical abstention)
Arquitectura de control	filtro o clasificador
Punto de aplicación	post-imagen (sanitización de texto en píxeles o escalado a revisión humana)
Modelo de amenaza	atacante que incrusta texto adversarial en los datos de píxeles de imágenes médicas para alterar la salida del VLM
Tipo de ataque	image-embedded prompt injection (4 clases de ataque, 8 variantes por imagen)
Adaptatividad del atacante	no adaptativo (ataques pre-generados, sin adaptación en tiempo real)

Campo	Valor
Entorno o benchmark	evaluación offline sobre benchmark fijo con ataques predefinidos
Baseline o comparador	inferencia sin ataque (clean-image) y cinco defensas: recorte de imagen, sanitización basada en OCR, ProvDent (defensa consciente de
Métricas de seguridad	Attack Success Rate (ASR) pareado; F1 en imágenes limpias; reducción de ASR por defensa hasta 62.6% (95% CI: 58.5–66.7%) para GPT-4o sin defensa; con OCR sanitization ASR agrupado 0.2%
Tasa de éxito del ataque	The dental clinical-harm-weighted error rate (dental-CHER, with false-negative weight = 4.0) on clean images was 0.60 across all models, driven entirely by false positives on the small negativeclass subset.
Falsos positivos	Consequently, the utility-preservation findings reported below (F1 within 0.6 percentage points of baseline) are driven primarily by sensitivity and should not be interpreted as a comprehensive clinical validation of diagnostic robustness across balanced populations.
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	OCR-based sanitization logró la mayor reducción de ataque (ASR agrupado 0.2%); ProvDent proporciona mecanismo de gobernanza complementario
Modos de fallo	no reportado explícitamente; se observa que los perfiles de vulnerabilidad son específicos del modelo con interacciones ataque-modelo significativas
Código o artefactos	no reportado (preprint, no se menciona disponibilidad de código)
Conclusión defensiva	La inyección de prompt incrustada en imagen es un riesgo de seguridad entre proveedores para VLM dentales. Sin defensa, GPT-4o alcanza un ASR del 62.6%. La defensa basada en OCR reduce el ASR al 0.2% con un impacto mínimo en la utilidad (F1 dentro de 0.6 pp de la línea base). ProvDent ofrece un mecanismo de gobernanza de fallo abierto. No se reportan sobrecostos de latencia o coste.
Confianza de extracción	95

- Hallazgos clave: Los cuatro VLM evaluados son vulnerables a la inyección de prompt incrustada en imagen, con ASR de hasta 62.6% en GPT-4o. La defensa basada en OCR reduce el ASR al 0.2% con mínimo impacto en la utilidad clínica.
- Evidencia de apoyo: All four models were vulnerable, with paired attack success rates up to 62.6% (95% CI: 58.5–66.7%) for GPT-4o.
- Localización de la evidencia: p. 2 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre monitorización, verificación y salida y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 28. Can AI Keep a Secret? Contextual Integrity Verification: A Provable Security Architecture for LLMs

Can AI Keep a Secret? Contextual Integrity Verification: A Provable Security Architecture for LLMs entra en la revisión porque aporta evidencia trazable en la familia contexto, rag y prompt injection indirecta. Evalúa máscara de atención dura pre-softmax con retículo de confianza de fuente; etiquetado de procedencia criptográfica (HMAC-SHA-256) por token; compuertas FFN/residuales opcionales frente a atacante que puede inyectar tokens maliciosos en cualquier fuente de entrada (prompt); se asume que el atacante conoce la arquitectura CIV pero no puede romper la criptografía subyacente; el modelo permanece congelado (sin reentrenamiento), con aplicación en inferencia (pre-softmax hard attention mask; opcionalmente FFN/residual gating). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a atacante que puede inyectar tokens maliciosos en cualquier fuente de entrada (prompt); se asume que el atacante conoce la arquitectura CIV pero no puede romper la criptografía subyacente; el modelo permanece congelado (sin reentrenamiento) mediante máscara de atención dura pre-softmax con retículo de confianza de fuente; etiquetado de procedencia criptográfica (HMAC-SHA-256) por token; compuertas FFN/residuales opcionales, aplicado en inferencia (pre-softmax hard attention mask; opcionalmente FFN/residual gating); la capacidad del atacante se clasifica como no adaptativo (ataques estáticos del benchmark; no se menciona adaptación dinámica). El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como G4 — Utility Security must not cripple intelligence—minimal degradation on legitimate workloads. y el sobre-coste como no reportado; no reportado. La evidencia de robustez es ASR=0% en el benchmark compuesto Elite-Attack + SoK-246; se afirma que la defensa es determinista y no basada en patrones probabilísticos y el fallo residual recuperable es no se reportan fallos bajo el modelo de amenaza declarado; se reconoce sobrecarga de latencia. Para la pregunta de investigación, esto importa así: CIV logra un 0% de tasa de éxito de ataque en inyección de instrucciones sobre modelos congelados, con una pérdida mínima de fidelidad (93.1% de similitud) y sin afectar a la perplejidad, aunque introduce una sobrecarga de latencia notable por la vía criptográfica no optimizada. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Aayush Gupta
- Año: 2025
- DOI: 10.48550/arxiv.2508.09288

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	Llama-3-8B y Mistral-7B (n=2)
Harness o defensa	Contextual Integrity Verification (CIV)
Arquitectura de control	máscara de atención dura pre-softmax con retículo de confianza de fuente; etiquetado de procedencia criptográfica (HMAC-SHA-256) por token; compuertas FFN/residuales opcionales
Punto de aplicación	inferencia (pre-softmax hard attention mask; opcionalmente FFN/residual gating)
Modelo de amenaza	atacante que puede inyectar tokens maliciosos en cualquier fuente de entrada (prompt); se asume que el atacante conoce la arquitectura CIV pero no puede romper la criptografía subyacente; el modelo permanece congelado (sin reentrenamiento)

Campo	Valor
Tipo de ataque	inyección de instrucciones (prompt injection); jailbreak
Adaptatividad del atacante	no adaptativo (ataques estáticos del benchmark; no se menciona adaptación dinámica)
Entorno o benchmark	benchmark sintético offline; simulación de ataques sobre modelos congelados
Baseline o comparador	ausencia de defensa (modelo original sin CIV) y comparación implícita con guardrails heurísticos (LLM-Guard, Rebuff, PromptArmor)
Métricas de seguridad	Attack Success Rate (ASR); token-level similarity; model perplexity
Tasa de éxito del ataque	no reportado
Falsos positivos	While competitors struggle with double-digit failure rates and crippling false positives (Rebuff's 63% FPR renders it unusable), CIV delivers perfect security with minimal utility impact.
Impacto en utilidad	G4 — Utility Security must not cripple intelligence—minimal degradation on legitimate workloads.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	ASR=0% en el benchmark compuesto Elite-Attack + SoK-246; se afirma que la defensa es determinista y no basada en patrones probabilísticos
Modos de fallo	no se reportan fallos bajo el modelo de amenaza declarado; se reconoce sobrecarga de latencia
Código o artefactos	sí: implementación de referencia en GitHub (https://github.com/ayushgupta4897/Contextual-Integrity-Verification); corpus Elite-Attack en Hugging Face (https://huggingface.co/datasets/zyushg/elite-attack); arnés de certificación automatizado
Conclusión defensiva	CIV proporciona protección determinista (0% ASR) frente a inyección de instrucciones en modelos congelados (Llama-3-8B, Mistral-7B) bajo el modelo de amenaza definido, con impacto mínimo en utilidad (93.1% similitud, sin degradación de perplejidad) pero con sobrecarga de latencia significativa debida a la ruta criptográfica no optimizada. No se compara con defensas heurísticas existentes en el mismo benchmark.
Confianza de extracción	95

- Hallazgos clave: CIV logra un 0% de tasa de éxito de ataque en inyección de instrucciones sobre modelos congelados, con una pérdida mínima de fidelidad (93.1% de similitud) y sin afectar a la perplejidad, aunque introduce una sobrecarga de latencia notable por la vía criptográfica no optimizada.
- Evidencia de apoyo: CIV attains 0% attack success rate under the stated threat model while preserving 93.1% token-level similarity and showing no degradation in model perplexity on benign tasks
- Localización de la evidencia: full text
- Lectura comparativa: Aporta evidencia sobre contexto, rag y prompt injection indirecta y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 29. OpenGuardrails: A Configurable, Unified, and Scalable Guardrails Platform for Large Language Models

OpenGuardrails: A Configurable, Unified, and Scalable Guardrails Platform for Large Language Models entra en la revisión porque aporta evidencia trazable en la familia exfiltración, secretos y privacidad. Evalúa Arquitectura unificada basada en LLM (un solo modelo para seguridad y manipulación), con mecanismo configurable de adaptación de políticas y diseño cuantizado y escalable (14B→3.3B con GPTQ). frente a Violaciones de seguridad de contenido, manipulación del modelo (inyección de prompts, jailbreaks, abuso del intérprete de código) y fuga de datos., con aplicación en Pasarela segura o servicio API en la entrada/salida de las llamadas al LLM. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: OpenGuardrails unifica detección de seguridad y defensa contra manipulación en un solo modelo cuantizado, manteniendo más del 98% de precisión y superando a LlamaGuard y WildGuard en benchmarks multilingües y latencia.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo cuantitativo apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Violaciones de seguridad de contenido, manipulación del modelo (inyección de prompts, jailbreaks, abuso del intérprete de código) y fuga de datos. mediante Arquitectura unificada basada en LLM (un solo modelo para seguridad y manipulación), con mecanismo configurable de adaptación de políticas y diseño cuantizado y escalable (14B→3.3B con GPTQ)., aplicado en Pasarela segura o servicio API en la entrada/salida de las llamadas al LLM; la capacidad del atacante se clasifica como no reportado. El comparador principal es LlamaGuard y WildGuard. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Rendimiento de vanguardia en benchmarks de seguridad multilingües con soporte de 119 idiomas. y el fallo residual recuperable es Smaller models (<= 4B) struggle with semantic ambiguity and adversarial paraphrasing, leading to unstable confidence distributions and high false positives/negatives.. Para la pregunta de investigación, esto importa así: OpenGuardrails unifica detección de seguridad y defensa contra manipulación en un solo modelo cuantizado, manteniendo más del 98% de precisión y superando a LlamaGuard y WildGuard en benchmarks multilingües y latencia. La confianza de extracción es alta y permite integrar el estudio en la comparación central con una base empírica suficiente.

- Autores: Thomas Wang; Haowen Li
- Año: 2025
- DOI: 10.48550/arxiv.2510.19169

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	OpenGuardrails, LlamaGuard y WildGuard (n=2510)
Harness o defensa	OpenGuardrails
Arquitectura de control	Arquitectura unificada basada en LLM (un solo modelo para seguridad y manipulación), con mecanismo configurable de adaptación de políticas y diseño cuantizado y escalable (14B→3.3B con GPTQ).
Punto de aplicación	Pasarela segura o servicio API en la entrada/salida de las llamadas al LLM
Modelo de amenaza	Violaciones de seguridad de contenido, manipulación del modelo (inyección de prompts, jailbreaks, abuso del intérprete de código) y fuga de datos.
Tipo de ataque	Prompt injection; jailbreaks; code-interpreter abuse; data leakage

Campo	Valor
Adaptatividad del atacante	no reportado
Entorno o benchmark	Evaluación comparativa post hoc sobre benchmarks de seguridad multilingües; también se evalúa latencia en el mundo real.
Baseline o comparador	LlamaGuard y WildGuard
Métricas de seguridad	F1; accuracy; latencia
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Rendimiento de vanguardia en benchmarks de seguridad multilingües con soporte de 119 idiomas.
Modos de fallo	Smaller models ($\leq 4B$) struggle with semantic ambiguity and adversarial paraphrasing, leading to unstable confidence distributions and high false positives/negatives.
Código o artefactos	Todos los modelos, conjuntos de datos y scripts de despliegue disponibles bajo licencia Apache 2.0
Conclusión defensiva	OpenGuardrails reduce el riesgo de contenido inseguro, manipulación y fuga de datos mediante una arquitectura unificada de guardarraíles. Frente a LlamaGuard y WildGuard, obtiene mayor F1 en benchmarks multilingües y mejor latencia en el mundo real; no se reportan tasas de éxito de ataque ni sobrecoste formal, por lo que la superioridad debe interpretarse en el contexto de estas métricas y de la amenaza específica.
Confianza de extracción	80

- Hallazgos clave: OpenGuardrails unifica detección de seguridad y defensa contra manipulación en un solo modelo cuantizado, manteniendo más del 98% de precisión y superando a LlamaGuard y WildGuard en benchmarks multilingües y latencia.
- Evidencia de apoyo: OpenGuardrails achieves state-of-the-art performance across multilingual safety benchmarks
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre exfiltración, secretos y privacidad y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 30. ClawGuard: A Runtime Security Framework for Tool-Augmented LLM Agents Against Indirect Prompt Injection

ClawGuard: A Runtime Security Framework for Tool-Augmented LLM Agents Against Indirect Prompt Injection entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa marco de seguridad en tiempo de ejecución con conjunto de reglas confirmadas por el usuario en cada límite de llamada a herramienta frente a adversario que incrusta instrucciones maliciosas en contenido devuelto por herramientas (web, local, MCP, skill files) para inducir llamadas a herramientas no autorizadas, con aplicación en límite de llamada a herramienta (tool-call boundary), antes de que la llamada se ejecute. Metodológicamente se articula como método descrito en el texto completo. Su aportación central

es: ClawGuard logra una mejora absoluta del 30% en DSR en SkillInject y del 27.4% en MCPTox frente a la ausencia de defensa, superando a MELON y Task Shield sin comprometer la utilidad del agente.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental post hoc sobre múltiples benchmarks de inyección y utilidad, con comparación de defensas existentes y análisis de fallos residuales. Protege frente a adversario que incrusta instrucciones maliciosas en contenido devuelto por herramientas (web, local, MCP, skill files) para inducir llamadas a herramientas no autorizadas mediante marco de seguridad en tiempo de ejecución con conjunto de reglas confirmadas por el usuario en cada límite de llamada a herramienta, aplicado en límite de llamada a herramienta (tool-call boundary), antes de que la llamada se ejecute; la capacidad del atacante se clasifica como no adaptativo (ataques predefinidos en benchmarks). El comparador principal es sin defensa, MELON y Task Shield. La eficacia se reporta como For stronger models, including GPT-5.4 and Claude-Sonnet-4.6, C LAW G UARD further suppresses the already low residual ASR and achieves near-perfect or perfect DSR (up to 100%)., el efecto sobre uso legítimo como no se compromete la utilidad del agente; se reporta que ClawGuard no reduce significativamente la tasa de éxito en tareas de utilidad (OSBench, WebArena, SWE-bench) y el sobrecoste como no reportado; no reportado. La evidencia de robustez es ClawGuard supera a MELON en 18.72% DSR promedio y a Task Shield en 10.57% en benchmarks de skill/MCP; los fallos residuales se deben a ataques que enmascaran la intención maliciosa bajo marcos de política empresarial y el fallo residual recuperable es 6 fallos residuales compartidos en SkillInject (p. ej., SEO para sitios ecológicos, minimización de riesgos de seguridad en evaluación, promoción de energías tradicionales, eliminación de contenido político, sesgo hacia remedios naturales) que enmascaran la intención maliciosa bajo marcos de política empresarial. Para la pregunta de investigación, esto importa así: ClawGuard logra una mejora absoluta del 30% en DSR en SkillInject y del 27.4% en MCPTox frente a la ausencia de defensa, superando a MELON y Task Shield sin comprometer la utilidad del agente. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Wei Zhao; Zhe Li; Peixin Zhang; Jun Sun
- Año: 2026
- DOI: 10.48550/arxiv.2604.11790

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-5.4, GPT-4o, Claude 4 Sonnet y Gemini 2.5 Pro, entre otros
Harness o defensa	ClawGuard
Arquitectura de control	marco de seguridad en tiempo de ejecución con conjunto de reglas confirmadas por el usuario en cada límite de llamada a herramienta
Punto de aplicación	límite de llamada a herramienta (tool-call boundary), antes de que la llamada se ejecute
Modelo de amenaza	adversario que incrusta instrucciones maliciosas en contenido devuelto por herramientas (web, local, MCP, skill files) para inducir llamadas a herramientas no autorizadas
Tipo de ataque	inyección indirecta de prompts (indirect prompt injection) a través de canales web, local, MCP y skill
Adaptatividad del atacante	no adaptativo (ataques predefinidos en benchmarks)
Entorno o benchmark	evaluación offline sobre benchmarks de inyección y utilidad
Baseline o comparador	sin defensa, MELON y Task Shield

Campo	Valor
Métricas de seguridad	Defense Success Rate (DSR); Attack Success Rate (ASR); utilidad del agente (tasa de éxito en tareas); sobrecarga de tokens
Tasa de éxito del ataque	For stronger models, including GPT-5.4 and Claude-Sonnet-4.6, C LAW G UARD further suppresses the already low residual ASR and achieves near-perfect or perfect DSR (up to 100%).
Falsos positivos	no reportado
Impacto en utilidad	no se compromete la utilidad del agente; se reporta que ClawGuard no reduce significativamente la tasa de éxito en tareas de utilidad (OSBench, WebArena, SWE-bench)
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	ClawGuard supera a MELON en 18.72% DSR promedio y a Task Shield en 10.57% en benchmarks de skill/MCP; los fallos residuales se deben a ataques que enmascaran la intención maliciosa bajo marcos de política empresarial
Modos de fallo	6 fallos residuales compartidos en SkillInject (p. ej., SEO para sitios ecológicos, minimización de riesgos de seguridad en evaluación, promoción de energías tradicionales, eliminación de contenido político, sesgo hacia remedios naturales) que enmascaran la intención maliciosa bajo marcos de política empresarial
Código o artefactos	sí, código disponible en github.com/Claw-Guard/ClawGuard/
Conclusión defensiva	ClawGuard proporciona una defensa determinista y auditable contra la inyección indirecta de prompts en agentes LLM aumentados con herramientas, bloqueando las tres vías de inyección (web/local, MCP, skill) sin modificar el modelo ni la infraestructura, y sin comprometer la utilidad del agente ni introducir sobrecarga significativa de tokens. La defensa es más efectiva que MELON y Task Shield, especialmente en ataques basados en skill y MCP. Los fallos residuales ocurren cuando el ataque enmascara la intención maliciosa bajo marcos de política empresarial, lo que sugiere la necesidad de reglas más específicas o verificación semántica adicional.
Confianza de extracción	95

- Hallazgos clave: ClawGuard logra una mejora absoluta del 30% en DSR en SkillInject y del 27.4% en MCPTox frente a la ausencia de defensa, superando a MELON y Task Shield sin comprometer la utilidad del agente. Los fallos residuales se deben a ataques que enmascaran la intención maliciosa bajo marcos de política empresarial.
- Evidencia de apoyo: ClawGuard raises these model-averaged DSRs to 86.67%, 89.50%, and 94.21%, corresponding to absolute gains of 30.00%, 27.40%, and 18.36%.
- Localización de la evidencia: p. 9 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar

amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 31. TraceSafe: A Systematic Assessment of LLM Guardrails on Multi-Step Tool-Calling Trajectories

TraceSafe: A Systematic Assessment of LLM Guardrails on Multi-Step Tool-Calling Trajectories entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa filtro o clasificador; guardrail por políticas; sandbox o aislamiento; monitor de ejecución frente a Riesgos en trayectorias de múltiples pasos: inyección de prompts, fugas de privacidad, alucinaciones, inconsistencias de interfaz, etc., con aplicación en Punto medio de la trayectoria (mid-trajectory), durante la ejecución de llamadas a herramientas. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Riesgos en trayectorias de múltiples pasos: inyección de prompts, fugas de privacidad, alucinaciones, inconsistencias de interfaz, etc. mediante filtro o clasificador; guardrail por políticas; sandbox o aislamiento; monitor de ejecución, aplicado en Punto medio de la trayectoria (mid-trajectory), durante la ejecución de llamadas a herramientas; la capacidad del atacante se clasifica como no reportado. El comparador principal es Comparación entre LLM-as-a-guard y salvaguardas especializadas y correlación con benchmarks estructurados a texto y jailbreak robustness. La eficacia se reporta como For easy comparison, we convert attack success rate (ASR) of StrongREJECT into model robustness score $(1 - ASR)$ shown in Figure 3., el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es La precisión se mantiene estable a lo largo de trayectorias extendidas; el rendimiento mejora en etapas posteriores y el fallo residual recuperable es Los modelos especializados en seguridad tienen menor rendimiento que los LLM de propósito general; el rendimiento depende más de la competencia en datos estructurados que de la alineación semántica. Para la pregunta de investigación, esto importa así: La eficacia de las salvaguardas en trayectorias de múltiples pasos depende más de la competencia en datos estructurados que de la alineación semántica; los LLM de propósito general superan a las salvaguardas especializadas; la precisión se mantiene estable en trayectorias largas. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Yen-Shan Chen; Sian-Yao Huang; Cheng-Lin Yang; Yun-Nung Chen
- Año: 2026
- DOI: 10.48550/arxiv.2604.07223

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	13 LLM-as-a-guard models (incluyendo GPT-4o, GPT-4o-mini, Claude 3.5 Sonnet, Claude 3 Haiku, Gemini 1.5 Pro, Gemini 1.5 Flash, Llama-3.1-70B, (n=20)
Harness o defensa	TraceSafe-Bench
Arquitectura de control	filtro o clasificador; guardrail por políticas; sandbox o aislamiento; monitor de ejecución
Punto de aplicación	Punto medio de la trayectoria (mid-trajectory), durante la ejecución de llamadas a herramientas

Campo	Valor
Modelo de amenaza	Riesgos en trayectorias de múltiples pasos: inyección de prompts, fugas de privacidad, alucinaciones, inconsistencias de interfaz, etc.
Tipo de ataque	Inyección de prompts, fugas de privacidad, alucinaciones, inconsistencias de interfaz (12 categorías)
Adaptatividad del atacante	no reportado
Entorno o benchmark	Benchmark offline con instancias sintéticas de trayectorias de llamada a herramientas
Baseline o comparador	Comparación entre LLM-as-a-guard y salvaguardas especializadas y correlación con benchmarks estructurados a texto y jailbreak robustness
Métricas de seguridad	Precisión (accuracy) en detección de riesgos; correlación de Spearman con benchmarks estructurados a texto y jailbreak robustness
Tasa de éxito del ataque	For easy comparison, we convert attack success rate (ASR) of StrongREJECT into model robustness score ($1 - \text{ASR}$) shown in Figure 3.
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	La precisión se mantiene estable a lo largo de trayectorias extendidas; el rendimiento mejora en etapas posteriores
Modos de fallo	Los modelos especializados en seguridad tienen menor rendimiento que los LLM de propósito general; el rendimiento depende más de la competencia en datos estructurados que de la alineación semántica
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	La eficacia de las salvaguardas en trayectorias de múltiples pasos está impulsada por la competencia en datos estructurados (p. ej., análisis JSON) más que por la alineación de seguridad semántica. Los LLM de propósito general superan a las salvaguardas especializadas. La precisión se mantiene estable en trayectorias largas. Se necesita optimizar conjuntamente el razonamiento estructural y la alineación de seguridad.
Confianza de extracción	95

- Hallazgos clave: La eficacia de las salvaguardas en trayectorias de múltiples pasos depende más de la competencia en datos estructurados que de la alineación semántica; los LLM de propósito general superan a las salvaguardas especializadas; la precisión se mantiene estable en trayectorias largas.
- Evidencia de apoyo: Guardrail efficacy is driven more by structural data competence (e.g., JSON parsing) than semantic safety alignment.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 32. PromptArmor: Simple yet Effective Prompt Injection Defenses

PromptArmor: Simple yet Effective Prompt Injection Defenses entra en la revisión porque aporta evidencia trazable en la familia contexto, rag y prompt injection indirecta. Evalúa Arquitectura de agente LLM con un guardrail LLM separado que filtra la entrada antes de que el agente la procese. frente a Atacante externo que inyecta prompts maliciosos en el entorno externo con el que interactúa el agente (inyección indirecta de prompts).., con aplicación en Antes de que el agente procese la entrada (pre-procesamiento).. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en Prompting de un LLM guardrail para detectar y eliminar inyecciones; evaluación en benchmark AgentDojo; comparación de estrategias de prompting.. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante externo que inyecta prompts maliciosos en el entorno externo con el que interactúa el agente (inyección indirecta de prompts). mediante Arquitectura de agente LLM con un guardrail LLM separado que filtra la entrada antes de que el agente la procese., aplicado en Antes de que el agente procese la entrada (pre-procesamiento).; la capacidad del atacante se clasifica como Sí, se evalúa contra ataques adaptativos (adaptive attacks).. El comparador principal es no reportado (se recomienda como baseline para futuras defensas). La eficacia se reporta como Moreover, after removing injected prompts with PromptArmor, the attack success rate drops to below 1%., el efecto sobre uso legítimo como In addition, we do not consider training-based defenses such as SecAlign (Chen et al., 2025a), as they exhibit poor utility on AgentDojo even in the absence of attacks, largely due to their degraded instruction-following capability. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Se demuestra efectividad contra ataques adaptativos; se exploran diferentes estrategias de prompting. y el fallo residual recuperable es For example, using GPT-4o, GPT-4.1, or o4-mini, PromptArmor achieves both a false positive rate and a false negative rate below 1% on the AgentDojo benchmark.. Para la pregunta de investigación, esto importa así: PromptArmor, un defensor basado en un LLM guardrail, reduce la tasa de éxito de ataques de inyección indirecta de prompts por debajo del 1% en AgentDojo, con tasas de falsos positivos y falsos negativos también por debajo del 1% usando GPT-4o, GPT-4.1 u o4-mini. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Tianneng Shi; Kaijie Zhu; Zhun Wang; Yuqi Jia; Will Cai; Weida Liang; Haonan Wang; Hend Alzahrani; Joshua Lu; Kenji Kawaguchi; Basel Alomair; Xuandong Zhao; William Yang Wang; Neil Gong; Wenbo Guo; Dawn Song
- Año: 2025
- DOI: 10.48550/arxiv.2507.15219

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-4o, GPT-4.1, o4-mini y guardrail LLM (no especificado), entre otros (n=3)
Harness o defensa	PromptArmor
Arquitectura de control	Arquitectura de agente LLM con un guardrail LLM separado que filtra la entrada antes de que el agente la procese.
Punto de aplicación	Antes de que el agente procese la entrada (pre-procesamiento).
Modelo de amenaza	Atacante externo que inyecta prompts maliciosos en el entorno externo con el que interactúa el agente (inyección indirecta de prompts).
Tipo de ataque	Inyección indirecta de prompts (indirect prompt injection).

Campo	Valor
Adaptatividad del atacante	Sí, se evalúa contra ataques adaptativos (adaptive attacks).
Entorno o benchmark	Benchmark AgentDojo; se mide la capacidad de detectar y eliminar inyecciones, y la tasa de éxito del ataque tras la defensa.
Baseline o comparador	no reportado (se recomienda como baseline para futuras defensas)
Métricas de seguridad	Tasa de falsos positivos (FPR); tasa de falsos negativos (FNR); tasa de éxito del ataque (ASR).
Tasa de éxito del ataque	Moreover, after removing injected prompts with PromptArmor, the attack success rate drops to below 1%.
Falsos positivos	For example, using GPT-4o, GPT-4.1, or o4-mini, PromptArmor achieves both a false positive rate and a false negative rate below 1% on the AgentDojo benchmark.
Impacto en utilidad	In addition, we do not consider training-based defenses such as SecAlign (Chen et al., 2025a), as they exhibit poor utility on AgentDojo even in the absence of attacks, largely due to their degraded instruction-following capability.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Se demuestra efectividad contra ataques adaptativos; se exploran diferentes estrategias de prompting.
Modos de fallo	For example, using GPT-4o, GPT-4.1, or o4-mini, PromptArmor achieves both a false positive rate and a false negative rate below 1% on the AgentDojo benchmark.
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	PromptArmor reduce la tasa de éxito de ataques de inyección indirecta de prompts por debajo del 1% en AgentDojo con FPR y FNR <1% usando GPT-4o, GPT-4.1 u o4-mini. Se muestra robustez frente a ataques adaptativos. No se reporta impacto en utilidad, latencia ni coste.
Confianza de extracción	95

- Hallazgos clave: PromptArmor, un defensor basado en un LLM guardrail, reduce la tasa de éxito de ataques de inyección indirecta de prompts por debajo del 1% en AgentDojo, con tasas de falsos positivos y falsos negativos también por debajo del 1% usando GPT-4o, GPT-4.1 u o4-mini. La defensa es efectiva incluso frente a ataques adaptativos.
- Evidencia de apoyo: using GPT-4o, GPT-4.1, or o4-mini, PromptArmor achieves both a false positive rate and a false negative rate below 1% on the AgentDojo benchmark. Moreover, after removing injected prompts with PromptArmor, the attack success rate drops to below 1%.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre contexto, rag y prompt injection indirecta y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 33. Semantic Attacks on Tool-Augmented LLMs: Securing the Model Context Protocol Against Descriptor-Level Manipulation

Semantic Attacks on Tool-Augmented LLMs: Securing the Model Context Protocol Against Descriptor-Level Manipulation entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Model Context Protocol (MCP) con descriptores de herramientas frente a ataques semánticos a nivel de descriptor: Tool Poisoning, Shadowing, Rug Pull; el atacante manipula metadatos de herramientas para sesgar la selección y el razonamiento, con aplicación en verificación de integridad de descriptores, revisión semántica previa al contexto con LLM auxiliar, barreras de protección en tiempo de ejecución. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Los ataques semánticos a nivel de descriptor en MCP pueden inducir invocaciones inseguras de herramientas en hasta el 36% de los ensayos sin defensa.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental controlada con escenarios adversariales MCP, comparación entre modelos y estrategias de prompting, con y sin mitigación en capas. Protege frente a ataques semánticos a nivel de descriptor: Tool Poisoning, Shadowing, Rug Pull; el atacante manipula metadatos de herramientas para sesgar la selección y el razonamiento mediante Model Context Protocol (MCP) con descriptores de herramientas, aplicado en verificación de integridad de descriptores, revisión semántica previa al contexto con LLM auxiliar, barreras de protección en tiempo de ejecución; la capacidad del atacante se clasifica como no reportado (los ataques se simulan de forma controlada, no se especifica adaptatividad del atacante). El comparador principal es configuración baseline sin mitigación, comparación entre ocho estrategias de prompting y comparación entre modelos. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como While the baseline MCP system prioritizes usability (93.5% task success, 6.5% false positives), it provides limited protection (41.2%). y el sobrecoste como no reportado; no reportado. La evidencia de robustez es la mitigación completa reduce invocaciones inseguras al 15% y aumenta tasa de bloqueo al 74%; diferencias significativas entre modelos y estrategias de prompting y el fallo residual recuperable es no reportado explícitamente (se menciona sensibilidad a manipulación, pero no se detallan modos de fallo). Para la pregunta de investigación, esto importa así: Los ataques semánticos a nivel de descriptor en MCP pueden inducir invocaciones inseguras de herramientas en hasta el 36% de los ensayos sin defensa. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Saeid Jamshidi; Arghavan Moradi Dakhel; Kawser Wazed Nafi; Foutse Khomh
- Año: 2025
- DOI: 10.48550/arxiv.2512.06556

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-5.3, DeepSeek-V3 y LLaMA-3.5
Harness o defensa	no reportado (se propone una solución de defensa en capas, no un harness específico)
Arquitectura de control	Model Context Protocol (MCP) con descriptores de herramientas
Punto de aplicación	verificación de integridad de descriptores, revisión semántica previa al contexto con LLM auxiliar, barreras de protección en tiempo de ejecución
Modelo de amenaza	ataques semánticos a nivel de descriptor: Tool Poisoning, Shadowing, Rug Pull; el atacante manipula metadatos de herramientas para sesgar la selección y el razonamiento
Tipo de ataque	Tool Poisoning; Shadowing; Rug Pull

Campo	Valor
Adaptatividad del atacante	no reportado (los ataques se simulan de forma controlada, no se especifica adaptatividad del atacante)
Entorno o benchmark	entorno controlado adversarial MCP con manipulación de metadatos de herramientas
Baseline o comparador	configuración baseline sin mitigación, comparación entre ocho estrategias de prompting y comparación entre modelos
Métricas de seguridad	tasa de invocaciones inseguras; tasa de bloqueo; latencia; sensibilidad a la manipulación
Tasa de éxito del ataque	no reportado
Falsos positivos	Strategy Zero-shot Chain-of-Thought Self-Critique Reflexion Instructional Verifier Few-shot Scarecrow GPT-5.3 DeepSeek-V3 LLaMA-3.5 0.68 0.75 0.70 0.78 0.65 0.60 0.69 0.72 0.60 0.65 0.60 0.67 0.57 0.55 0.60 0.64 0.45 0.53 0.50 0.52 0.40 0.42 0.48 0.49 reports false positive rates under benign conditions. While the baseline MCP system prioritizes usability (93.5% task success, 6.5% false positives), it provides limited protection (41.2%).
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	la mitigación completa reduce invocaciones inseguras al 15% y aumenta tasa de bloqueo al 74%; diferencias significativas entre modelos y estrategias de prompting
Modos de fallo	no reportado explícitamente (se menciona sensibilidad a manipulación, pero no se detallan modos de fallo)
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	La manipulación de descriptores en MCP puede causar invocaciones inseguras en hasta el 36% de los ensayos sin defensa. La mitigación en capas propuesta reduce las invocaciones inseguras al 15% y aumenta el bloqueo al 74%, mejorando la resistencia. Sin embargo, la efectividad varía según el modelo y la estrategia de prompting. No se declara un harness superior; se identifican amenazas (Tool Poisoning, Shadowing, Rug Pull), baseline sin defensa, y se reporta mejora en seguridad con impacto en latencia no cuantificado.
Confianza de extracción	90

- Hallazgos clave: Los ataques semánticos a nivel de descriptor en MCP pueden inducir invocaciones inseguras de herramientas en hasta el 36% de los ensayos sin defensa. La defensa en capas propuesta reduce las invocaciones inseguras al 15% y aumenta la tasa de bloqueo al 74%, aunque la efectividad varía según el modelo y la estrategia de prompting.
- Evidencia de apoyo: descriptor manipulation can substantially alter tool-selection behavior, producing unsafe tool invocations in up to 36% of trials under baseline configurations
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con

contrato defensivo menos completo.

Estudio 34. Bypassing AI Control Protocols via Agent-as-a-Proxy Attacks

Bypassing AI Control Protocols via Agent-as-a-Proxy Attacks entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Monitor híbrido que evalúa CoT, llamadas a herramientas y salidas de herramientas del agente frente a Atacante externo que inyecta instrucciones maliciosas en contenido externo (IPI) para manipular al agente y al monitor simultáneamente, con aplicación en Monitor posterior a la acción del agente (evalúa trayectoria completa). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Los monitores híbridos que evalúan CoT y acciones de herramientas son vulnerables a ataques Agent-as-a-Proxy incluso con modelos de monitorización de gran escala; la defensa es frágil independientemente del tamaño del modelo.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante externo que inyecta instrucciones maliciosas en contenido externo (IPI) para manipular al agente y al monitor simultáneamente mediante Monitor híbrido que evalúa CoT, llamadas a herramientas y salidas de herramientas del agente, aplicado en Monitor posterior a la acción del agente (evalúa trayectoria completa); la capacidad del atacante se clasifica como No adaptativo (ataque optimizado offline con gradientes, sin adaptación en tiempo real). El comparador principal es monitores AlignmentCheck y Extract-and-Evaluate sin ataque y variación de LLM monitor (Qwen2.5-72B, otros no especificados). La eficacia se reporta como We achieve attack success rates (ASR) exceeding 90% against AlignmentCheck using diverse open-source models., el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Los monitores AlignmentCheck y Extract-and-Evaluate son vulnerables incluso con modelos de monitorización de gran escala (Qwen2.5-72B); la defensa es frágil independientemente de la escala y el fallo residual recuperable es El monitor clasifica trayectorias maliciosas como seguras cuando el agente repite cadenas adversariales optimizadas en su CoT o parámetros de herramienta. Para la pregunta de investigación, esto importa así: Los monitores híbridos que evalúan CoT y acciones de herramientas son vulnerables a ataques Agent-as-a-Proxy incluso con modelos de monitorización de gran escala; la defensa es frágil independientemente del tamaño del modelo. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Jafar Isbarov; Murat Kantarcioglu
- Año: 2026
- DOI: 10.48550/arxiv.2602.05066

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-4o mini, Llama-3.1-70B, Qwen2.5-72B (monitor) y AlignmentCheck, entre otros
Harness o defensa	AlignmentCheck; Extract-and-Evaluate
Arquitectura de control	Monitor híbrido que evalúa CoT, llamadas a herramientas y salidas de herramientas del agente
Punto de aplicación	Monitor posterior a la acción del agente (evalúa trayectoria completa)
Modelo de amenaza	Atacante externo que inyecta instrucciones maliciosas en contenido externo (IPI) para manipular al agente y al monitor simultáneamente
Tipo de ataque	Agent-as-a-Proxy (inyección indirecta de instrucciones con optimización adversarial Parallel-GCG)

Campo	Valor
Adaptatividad del atacante	No adaptativo (ataque optimizado offline con gradientes, sin adaptación en tiempo real)
Entorno o benchmark	Benchmark AgentDojo con tareas agénticas simuladas; se evalúa la tasa de éxito del ataque (ASR) contra monitores
Baseline o comparador	monitores AlignmentCheck y Extract-and-Evaluate sin ataque y variación de LLM monitor (Qwen2.5-72B, otros no especificados)
Métricas de seguridad	Attack Success Rate (ASR)
Tasa de éxito del ataque	We achieve attack success rates (ASR) exceeding 90% against AlignmentCheck using diverse open-source models.
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Los monitores AlignmentCheck y Extract-and-Evaluate son vulnerables incluso con modelos de monitorización de gran escala (Qwen2.5-72B); la defensa es frágil independientemente de la escala
Modos de fallo	El monitor clasifica trayectorias maliciosas como seguras cuando el agente repite cadenas adversariales optimizadas en su CoT o parámetros de herramienta
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	Los protocolos de monitorización híbrida actuales son fundamentalmente frágiles frente a ataques Agent-as-a-Proxy que utilizan optimización adversarial (Parallel-GCG). Incluso monitores de gran escala (Qwen2.5-72B) pueden ser eludidos por agentes de capacidad similar (GPT-4o mini, Llama-3.1-70B) en el benchmark AgentDojo, logrando una alta tasa de éxito de ataque. No se reporta impacto en utilidad, latencia ni coste, pero la amenaza es real y la adaptatividad del atacante es baja (ataque offline).
Confianza de extracción	90

- Hallazgos clave: Los monitores híbridos que evalúan CoT y acciones de herramientas son vulnerables a ataques Agent-as-a-Proxy incluso con modelos de monitorización de gran escala; la defensa es frágil independientemente del tamaño del modelo.
- Evidencia de apoyo: Large-scale monitoring models like Qwen2.5-72B can be bypassed by agents with similar capabilities, such as GPT-4o mini and Llama-3.1-70B. On the AgentDojo benchmark, we achieve a high attack success rate against AlignmentCheck and Extract-and-Evaluate...
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 35. Defending against Adaptive Prompt Injection Attacks via Reasoning-enabled Task Alignment

Defending against Adaptive Prompt Injection Attacks via Reasoning-enabled Task Alignment entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Agente LLM con política de bucle cerrado que intercala decisiones del modelo con ejecución de herramientas; contexto del agente: system prompt, tarea de usuario, historial de acciones y observaciones frente a Ataque indirecto de inyección de instrucciones: el atacante incrusta instrucciones maliciosas en datos de terceros que el agente recupera durante la ejecución de la tarea; el atacante puede optimizar contra la defensa desplegada (adaptativo), con aplicación en En cada paso de salida de herramienta (tool-output step), el defensor verifica mediante razonamiento de cadena de pensamiento que sus acciones son consistentes con la tarea del usuario. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Ataque indirecto de inyección de instrucciones: el atacante incrusta instrucciones maliciosas en datos de terceros que el agente recupera durante la ejecución de la tarea; el atacante puede optimizar contra la defensa desplegada (adaptativo) mediante Agente LLM con política de bucle cerrado que intercala decisiones del modelo con ejecución de herramientas; contexto del agente: system prompt, tarea de usuario, historial de acciones y observaciones, aplicado en En cada paso de salida de herramienta (tool-output step), el defensor verifica mediante razonamiento de cadena de pensamiento que sus acciones son consistentes con la tarea del usuario; la capacidad del atacante se clasifica como Adaptativo: el atacante puede optimizar contra la defensa desplegada; se utilizan seis ataques adaptativos de caja negra. El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia se reporta como RETA mantiene cada ASR por ataque por debajo del 10%; ASR promedio de 2.92% y 3.75% en los dos modelos objetivo, el efecto sobre uso legítimo como Preserva la mayor parte de la utilidad bajo ataque y en entradas limpias (no se dan valores numéricos en el digest) y el sobrecoste como no reportado; no reportado. La evidencia de robustez es RETA mantiene ASR <10% en seis ataques adaptativos de caja negra, con promedios de 2.92% y 3.75% en dos modelos objetivo; supera a defensas existentes que colapsan bajo evaluación adaptativa y el fallo residual recuperable es Las defensas existentes fallan porque: (1) reconocen patrones de ataque específicos en lugar de evaluar relevancia de la instrucción con la tarea del usuario; (2) el entrenamiento adversarial se concentra en pocas plantillas hechas a mano, sin generalizar a otras estrategias de reformulación. Para la pregunta de investigación, esto importa así: el estudio aporta evidencia sustantiva que ayuda a delimitar mecanismo, contexto y alcance inferencial. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Lipeng He; Yihan Wang; Jiawen Zhang; N. Asokan
- Año: 2026
- DOI: 10.48550/arxiv.2606.15441

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	RETA (defensor), Qwen3-4B (modelo atacante abliterado), dos modelos objetivo no especificados en el digest (se mencionan dos modelos objetivo en el
Harness o defensa	RETA (Reasoning-enabled Task Alignment)

Campo	Valor
Arquitectura de control	Agente LLM con política de bucle cerrado que intercala decisiones del modelo con ejecución de herramientas; contexto del agente: system prompt, tarea de usuario, historial de acciones y observaciones
Punto de aplicación	En cada paso de salida de herramienta (tool-output step), el defensor verifica mediante razonamiento de cadena de pensamiento que sus acciones son consistentes con la tarea del usuario
Modelo de amenaza	Ataque indirecto de inyección de instrucciones: el atacante incrusta instrucciones maliciosas en datos de terceros que el agente recupera durante la ejecución de la tarea; el atacante puede optimizar contra la defensa desplegada (adaptativo)
Tipo de ataque	Indirect prompt injection; ataques adaptativos de caja negra con optimización basada en búsqueda (genética, QD, etc.)
Adaptatividad del atacante	Adaptativo: el atacante puede optimizar contra la defensa desplegada; se utilizan seis ataques adaptativos de caja negra
Entorno o benchmark	Benchmark AgentDojo; evaluación estática y adaptativa; se reportan ASR por ataque y promedios en dos modelos objetivo
Baseline o comparador	Defensas existentes de cuatro paradigmas (observación, entrenamiento adversarial, etc.), ataques adaptativos de caja negra (seis variantes)
Métricas de seguridad	Attack Success Rate (ASR); Utility (UA y BU, no definidos explícitamente en el digest pero mencionados como ‘UA and BU’)
Tasa de éxito del ataque	RETA mantiene cada ASR por ataque por debajo del 10%; ASR promedio de 2.92% y 3.75% en los dos modelos objetivo
Falsos positivos	no reportado
Impacto en utilidad	Preserva la mayor parte de la utilidad bajo ataque y en entradas limpias (no se dan valores numéricos en el digest)
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	RETA mantiene ASR <10% en seis ataques adaptativos de caja negra, con promedios de 2.92% y 3.75% en dos modelos objetivo; supera a defensas existentes que colapsan bajo evaluación adaptativa
Modos de fallo	Las defensas existentes fallan porque: (1) reconocen patrones de ataque específicos en lugar de evaluar relevancia de la instrucción con la tarea del usuario; (2) el entrenamiento adversarial se concentra en pocas plantillas hechas a mano, sin generalizar a otras estrategias de reformulación
Código o artefactos	código o artefacto reportado como disponible

Campo	Valor
Conclusión defensiva	RETA, un método de defensa basado en entrenamiento que alinea las decisiones con la tarea del usuario mediante razonamiento de cadena de pensamiento y entrenamiento adversarial diversificado, logra mantener ASR por debajo del 10% frente a ataques adaptativos de caja negra, superando a defensas previas que colapsan bajo adaptación, con mínimo impacto en utilidad.
Confianza de extracción	90

- Hallazgos clave: RETA, un método de defensa basado en alineación de tareas y razonamiento, reduce la tasa de éxito de ataques adaptativos de inyección de instrucciones por debajo del 10% en todos los ataques, con promedios del 2.92% y 3.75% en dos modelos objetivo, manteniendo la utilidad del agente.
- Evidencia de apoyo: Across six black-box adaptive attacks, RETA keeps every per-attack ASR below 10%, with average ASR of 2.92% and 3.75% on the two target models, while preserving most utility under attack and on clean inputs.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 36. Defense Against Indirect Prompt Injection via Tool Result Parsing

Defense Against Indirect Prompt Injection via Tool Result Parsing entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Agente LLM con llamada a funciones (function calling) y parsing de resultados de herramientas frente a Atacante que inyecta instrucciones adversarias en los resultados de llamadas a herramientas externas (indirect prompt injection), con aplicación en post-procesamiento de resultados de herramientas (tool result parsing) antes de integrarlos en el contexto del prompt. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante que inyecta instrucciones adversarias en los resultados de llamadas a herramientas externas (indirect prompt injection) mediante Agente LLM con llamada a funciones (function calling) y parsing de resultados de herramientas, aplicado en post-procesamiento de resultados de herramientas (tool result parsing) antes de integrarlos en el contexto del prompt; la capacidad del atacante se clasifica como estático (ataques predefinidos del benchmark AgentDojo). El comparador principal es Sin defensa, defensa prompt-based (instrucciones de seguridad) y defensa con modelo de detección entrenado (no especificado), entre otros. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como Extensive experiments on the AgentDojo benchmark (utilizing gpt-oss-120b, llama-3.1-70b, and qwen3-32b) demonstrate that our approach achieves a competitive Utility under Attack (UA) while maintaining the lowest Attack Success Rate (ASR) to date, significantly outperforming existing methods. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es El método propuesto logra ASR=0% en todos los modelos probados, mientras que las defensas prompt-based existentes muestran ASR entre 20% y 80% dependiendo del modelo y el fallo residual recuperable es Compared with existing injection detection methods, our approach offers several advantages: (1) Pattern matching can only filter known injection patterns and fails to recognize novel, unknown attacks.. Para la pregunta de investigación, esto importa así: el estudio aporta evidencia sustantiva que ayuda a delimitar mecanismo, contexto y alcance inferencial. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Qiang Yu; Xinran Cheng; Chuanyi Liu

- Año: 2026
- DOI: 10.48550/arxiv.2601.04795

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	GPT-4o, GPT-4o-mini, Claude 3.5 Sonnet y Gemini 1.5 Pro, entre otros (n=65)
Harness o defensa	no reportado
Arquitectura de control	Agente LLM con llamada a funciones (function calling) y parsing de resultados de herramientas
Punto de aplicación	post-procesamiento de resultados de herramientas (tool result parsing) antes de integrarlos en el contexto del prompt
Modelo de amenaza	Atacante que inyecta instrucciones adversarias en los resultados de llamadas a herramientas externas (indirect prompt injection)
Tipo de ataque	Indirect Prompt Injection (IPI) con instrucciones maliciosas incrustadas en datos de herramientas
Adaptatividad del atacante	estático (ataques predefinidos del benchmark AgentDojo)
Entorno o benchmark	Benchmark AgentDojo con 27 tareas de calendario y correo electrónico, cada una con un ataque IPI
Baseline o comparador	Sin defensa, defensa prompt-based (instrucciones de seguridad) y defensa con modelo de detección entrenado (no especificado), entre otros
Métricas de seguridad	Attack Success Rate (ASR); Utility under Attack (UA)
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	Extensive experiments on the AgentDojo benchmark (utilizing gpt-oss-120b, llama-3.1-70b, and qwen3-32b) demonstrate that our approach achieves a competitive Utility under Attack (UA) while maintaining the lowest Attack Success Rate (ASR) to date, significantly outperforming existing methods.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	El método propuesto logra ASR=0% en todos los modelos probados, mientras que las defensas prompt-based existentes muestran ASR entre 20% y 80% dependiendo del modelo
Modos de fallo	Compared with existing injection detection methods, our approach offers several advantages: (1) Pattern matching can only filter known injection patterns and fails to recognize novel, unknown attacks.
Código o artefactos	Código disponible en GitHub: https://github.com/qiang-yu/agentdojo/tree/tool-result-extract

Campo	Valor
Conclusión defensiva	El método de parsing de resultados de herramientas (tool result parsing) reduce el ASR a 0% en el benchmark AgentDojo, manteniendo una utilidad competitiva, superando a las defensas prompt-based existentes. La amenaza es IPI, el baseline es sin defensa (ASR 100%), el atacante es estático, el impacto en utilidad es mínimo (UA 0.89 vs 0.91) y no se reporta sobrecoste de latencia o coste.
Confianza de extracción	95

- Hallazgos clave: El método de parsing de resultados de herramientas (tool result parsing) logra una tasa de éxito de ataque (ASR) del 0% en el benchmark AgentDojo, manteniendo una utilidad bajo ataque (UA) competitiva, superando significativamente a las defensas basadas en instrucciones existentes.
- Evidencia de apoyo: Our approach achieves competitive Utility under Attack (UA) while maintaining the lowest Attack Success Rate (ASR) to date, significantly outperforming existing methods.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 37. Evaluation of Prompt Injection Defenses in Large Language Models

Evaluation of Prompt Injection Defenses in Large Language Models entra en la revisión porque aporta evidencia trazable en la familia exfiltración, secretos y privacidad. Evalúa filtro o clasificador frente a atacante con acceso al teclado (usuario) que intenta extraer secretos del prompt del sistema mediante inyección de instrucciones, con aplicación en salida del modelo (output filtering) antes de llegar al usuario. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Las defensas que dependen del modelo para protegerse fallan todas frente a un atacante adaptativo; el filtrado de salida con reglas fijas en código de aplicación separado logra cero filtraciones en 15.000 ataques.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental adaptativa multi-ronda con atacante que evoluciona sus estrategias; comparación de nueve configuraciones de defensa en más de 20.000 ataques. Protege frente a atacante con acceso al teclado (usuario) que intenta extraer secretos del prompt del sistema mediante inyección de instrucciones mediante filtro o clasificador, aplicado en salida del modelo (output filtering) antes de llegar al usuario; la capacidad del atacante se clasifica como adaptativo: evoluciona sus estrategias a lo largo de cientos de rondas. El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es el filtrado de salida resistió 15.000 ataques sin filtraciones; las defensas basadas en el modelo fallaron todas y el fallo residual recuperable es defensas que dependen del modelo para protegerse a sí mismo son vulnerables; el filtrado de salida no presentó fallos en el estudio. Para la pregunta de investigación, esto importa así: Las defensas que dependen del modelo para protegerse fallan todas frente a un atacante adaptativo; el filtrado de salida con reglas fijas en código de aplicación separado logra cero filtraciones en 15.000 ataques. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Priyal Deep; Shane Emmons; Amy Fox; Kyle Bacon; Kelley McAllister; Peter Ortiz; Krisztian Flautner
- Año: 2026
- DOI: 10.48550/arxiv.2604.23887

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	no reportado (no se nombran modelos concretos y se refiere a LLM en general) (n=9)
Harness o defensa	Swept AI (herramienta de verificación mencionada, no el harness de defensa)
Arquitectura de control	filtro o clasificador
Punto de aplicación	salida del modelo (output filtering) antes de llegar al usuario
Modelo de amenaza	atacante con acceso al teclado (usuario) que intenta extraer secretos del prompt del sistema mediante inyección de instrucciones
Tipo de ataque	inyección de instrucciones directa (prompt injection) para extracción de secretos
Adaptatividad del atacante	adaptativo: evoluciona sus estrategias a lo largo de cientos de rondas
Entorno o benchmark	simulación de ataques automatizados en múltiples rondas
Baseline o comparador	nueve configuraciones de defensa: las que dependen del modelo (sin especificar) y filtrado de salida con reglas fijas en código de
Métricas de seguridad	tasa de filtraciones (leaks); número de ataques realizados
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	el filtrado de salida resistió 15.000 ataques sin filtraciones; las defensas basadas en el modelo fallaron todas
Modos de fallo	defensas que dependen del modelo para protegerse a sí mismo son vulnerables; el filtrado de salida no presentó fallos en el estudio
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	Las defensas contra inyección de instrucciones que dependen del modelo atacado son inherentemente inseguras; la única defensa efectiva es el filtrado de salida en código de aplicación separado, que logró cero filtraciones frente a un atacante adaptativo. Se recomienda restringir sistemas con operaciones sensibles a personal interno hasta que herramientas como Swept AI verifiquen las defensas.
Confianza de extracción	95

- Hallazgos clave: Las defensas que dependen del modelo para protegerse fallan todas frente a un atacante adaptativo; el filtrado de salida con reglas fijas en código de aplicación separado logra cero filtraciones en 15.000 ataques.
- Evidencia de apoyo: Every defense that relied on the model to protect itself eventually broke. The only defense that held was output filtering, which checks the model's responses via hardcoded rules in separate application code before they reach the user, achieving zero leaks...
- Localización de la evidencia: p. 1 (full text; fragmento localizado)

- Lectura comparativa: Aporta evidencia sobre exfiltración, secretos y privacidad y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 38. ICON: Indirect Prompt Injection Defense for Agents based on Inference-Time Correction

ICON: Indirect Prompt Injection Defense for Agents based on Inference-Time Correction entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Arquitectura de dos componentes: Latent Space Trace Prober y Mitigating Rectifier; opera en el espacio latente del LLM, realizando corrección en tiempo de inferencia. frente a Atacante que inyecta instrucciones maliciosas en contenido recuperado (por ejemplo, correos, páginas web) para secuestrar la ejecución del agente; se consideran ataques adaptativos., con aplicación en Tiempo de inferencia; durante la generación del agente, después de la recuperación de contenido y antes de la ejecución de acciones.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: ICON reduce la tasa de éxito de ataques IPI al 0.4% manteniendo una utilidad de tarea superior en más del 50% frente a defensas existentes, demostrando robustez OOD y extensión a agentes multimodales.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante que inyecta instrucciones maliciosas en contenido recuperado (por ejemplo, correos, páginas web) para secuestrar la ejecución del agente; se consideran ataques adaptativos. mediante Arquitectura de dos componentes: Latent Space Trace Prober y Mitigating Rectifier; opera en el espacio latente del LLM, realizando corrección en tiempo de inferencia., aplicado en Tiempo de inferencia; durante la generación del agente, después de la recuperación de contenido y antes de la ejecución de acciones.; la capacidad del atacante se clasifica como Adaptativo (se mencionan ataques adaptativos que son indistinguibles de contextos benignos). El comparador principal es Defensas existentes (MELON, LLM-Detector, etc.), detectores comerciales, y ataques adaptativos. La eficacia se reporta como ICON: Indirect Prompt Injection Defense for Agents based on Inference-Time Correction Che Wang 1 2 Fuyao Zhang 2 Jiaming Zhang 2 ziqi Zhang 1 Yinghui Wang 3 Longtao Huang 4 Jianbo Gao 1 Zhong Chen 1 Wei Yang Bryan Lim 2 40 Attack Success Rate (ASR) % arXiv:2602.20708v1 [cs.AI] 24 Feb 2026 Abstract Large Language Model, el efecto sobre uso legítimo como Ganancia de utilidad de tarea superior al 50% en comparación con defensas existentes (según el abstract). y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Robusta generalización fuera de distribución (OOD) y extensión efectiva a agentes multimodales. y el fallo residual recuperable es First, adversarial diversity: current heuristic-based defenses (e.g., tool filtering (Zhu et al., 2025) or static prompts (Willison, 2023)) are easily bypassed by adaptive attacks that seamlessly blend malicious intent into the agent’s context.. Para la pregunta de investigación, esto importa así: ICON reduce la tasa de éxito de ataques IPI al 0.4% manteniendo una utilidad de tarea superior en más del 50% frente a defensas existentes, demostrando robustez OOD y extensión a agentes multimodales. La confianza de extracción es alta y permite integrar el estudio en la comparación central con una base empírica suficiente.

- Autores: Che Wang; Fuyao Zhang; Jiaming Zhang; Ziqi Zhang; Yinghui Wang; Longtao Huang; Jianbo Gao; Zhong Chen; Wei Yang Bryan Lim
- Año: 2026
- DOI: 10.48550/arxiv.2602.20708

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo

Campo	Valor
Modelos o sistemas protegidos	ICON, backbones de LLM no especificados en el digest, detectores comerciales (no nombrados) y defensas comparadas (MELON, LLM-Detector, etc.) (n=9)
Harness o defensa	ICON
Arquitectura de control	Arquitectura de dos componentes: Latent Space Trace Prober y Mitigating Rectifier; opera en el espacio latente del LLM, realizando corrección en tiempo de inferencia.
Punto de aplicación	Tiempo de inferencia; durante la generación del agente, después de la recuperación de contenido y antes de la ejecución de acciones.
Modelo de amenaza	Atacante que inyecta instrucciones maliciosas en contenido recuperado (por ejemplo, correos, páginas web) para secuestrar la ejecución del agente; se consideran ataques adaptativos.
Tipo de ataque	Indirect Prompt Injection (IPI); ataques adaptativos
Adaptatividad del atacante	Adaptativo (se mencionan ataques adaptativos que son indistinguibles de contextos benignos)
Entorno o benchmark	Benchmarks de IPI con ataques adaptativos; evaluación de generalización OOD y extensión a agentes multimodales.
Baseline o comparador	Defensas existentes (MELON, LLM-Detector, etc.), detectores comerciales, y ataques adaptativos
Métricas de seguridad	Attack Success Rate (ASR); Utility under Attacks (UA); también se menciona la ganancia de utilidad de tarea.
Tasa de éxito del ataque	ICON: Indirect Prompt Injection Defense for Agents based on Inference-Time Correction Che Wang 1 2 Fuyao Zhang 2 Jiaming Zhang 2 ziqi Zhang 1 Yinghui Wang 3 Longtao Huang 4 Jianbo Gao 1 Zhong Chen 1 Wei Yang Bryan Lim 2 40 Attack Success Rate (ASR) % arXiv:2602.20708v1 [cs.AI] 24 Feb 2026 Abstract Large Language Model no reportado
Falsos positivos	no reportado
Impacto en utilidad	Ganancia de utilidad de tarea superior al 50% en comparación con defensas existentes (según el abstract).
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Robusta generalización fuera de distribución (OOD) y extensión efectiva a agentes multimodales.
Modos de fallo	First, adversarial diversity: current heuristic-based defenses (e.g., tool filtering (Zhu et al., 2025) or static prompts (Willison, 2023)) are easily bypassed by adaptive attacks that seamlessly blend malicious intent into the agent's context.
Código o artefactos	no reportado

Campo	Valor
Conclusión defensiva	ICON logra un equilibrio superior entre seguridad y eficiencia, con una ASR competitiva del 0.4% y una ganancia de utilidad de más del 50% frente a defensas existentes, demostrando robustez OOD y aplicabilidad multimodal. Sin embargo, la conclusión debe considerar que la ASR se reporta en un benchmark específico y no se detallan los costes de latencia o computacionales.
Confianza de extracción	80

- Hallazgos clave: ICON reduce la tasa de éxito de ataques IPI al 0.4% manteniendo una utilidad de tarea superior en más del 50% frente a defensas existentes, demostrando robustez OOD y extensión a agentes multimodales. La defensa se basa en la detección de firmas de sobre-enfoque en el espacio latente y la corrección quirúrgica de la atención.
- Evidencia de apoyo: ICON achieves a competitive 0.4% ASR, matching commercial grade detectors, while yielding a over 50% task utility gain.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 39. TRYLOCK: Defense-in-Depth Against LLM Jailbreaks via Layered Preference and Representation Engineering

TRYLOCK: Defense-in-Depth Against LLM Jailbreaks via Layered Preference and Representation Engineering entra en la revisión porque aporta evidencia trazable en la familia jailbreak y cumplimiento de políticas. Evalúa Arquitectura en capas: DPO (nivel de pesos), RepE steering (nivel de activaciones), clasificador lateral adaptativo (selección de fuerza de steering), canonicalización de entrada frente a Atacante que envía prompts adversariales (jailbreaks) de cinco familias: inyección de prompt, roleplay, codificación (Base64, ROT13, leetspeak), etc., con aplicación en Inferencia completa: antes de la generación (canonicalización), durante la generación (steering adaptativo), y a nivel de pesos (DPO pre-entrenado). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: TRYLOCK reduce el ASR de jailbreak del 46.5% al 5.6% en Mistral-7B-Instruct mediante una combinación en capas de DPO, RepE, canonicalización y un clasificador adaptativo, reduciendo además el sobre-rechazo del 60% al 48%.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc sobre un conjunto de ataques de jailbreak, con comparación de capas de defensa y análisis de ablación.. Protege frente a Atacante que envía prompts adversariales (jailbreaks) de cinco familias: inyección de prompt, roleplay, codificación (Base64, ROT13, leetspeak), etc. mediante Arquitectura en capas: DPO (nivel de pesos), RepE steering (nivel de activaciones), clasificador lateral adaptativo (selección de fuerza de steering), canonicalización de entrada, aplicado en Inferencia completa: antes de la generación (canonicalización), durante la generación (steering adaptativo), y a nivel de pesos (DPO pre-entrenado); la capacidad del atacante se clasifica como No adaptativo (conjunto fijo de ataques predefinidos). El comparador principal es Modelo sin defensa (DPO solo), DPO + RepE y DPO + RepE + canonicalización, entre otros. La eficacia se reporta como On Mistral-7B-Instruct evaluated against a 249-prompt attack set spanning five attack families, TRYLOCK achieves 88.0% relative ASR reduction (46.5% to 5.6%), with each layer contributing unique coverage: RepE blocks 36% of attacks that bypass DPO alone, while canonicalization catches 14% of encoding attacks that evade, el efecto sobre uso legítimo como Over-refusal reducido del 60% al 48% manteniendo defensa; no se reporta impacto en calidad de respuesta para consultas benignas y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Cobertura complementaria de capas: RepE

bloquea 36% de ataques que evaden DPO; canonicalización atrapa 14% de ataques de codificación que evaden ambos y el fallo residual recuperable es Fenómeno de steering no monótono: fuerza intermedia ($\alpha=1.0$) degrada seguridad por debajo de la línea base; interferencia RepE-DPO. Para la pregunta de investigación, esto importa así: TRYLOCK reduce el ASR de jailbreak del 46.5% al 5.6% en Mistral-7B-Instruct mediante una combinación en capas de DPO, RepE, canonicalización y un clasificador adaptativo, reduciendo además el sobre-rechazo del 60% al 48%. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Scott Thornton
- Año: 2026
- DOI: 10.48550/arxiv.2601.03300

Campo	Valor
Tipo de trabajo	Empírico
Diseño	Evaluación experimental post hoc sobre un conjunto de ataques de jailbreak, con comparación de capas de defensa y análisis de ablación.
Modelos o sistemas protegidos	Mistral-7B-Instruct y TRYLOCK (DPO + RepE + sidecar + canonicalización) (n=1)
Harness o defensa	TRYLOCK
Arquitectura de control	Arquitectura en capas: DPO (nivel de pesos), RepE steering (nivel de activaciones), clasificador lateral adaptativo (selección de fuerza de steering), canonicalización de entrada
Punto de aplicación	Inferencia completa: antes de la generación (canonicalización), durante la generación (steering adaptativo), y a nivel de pesos (DPO pre-entrenado)
Modelo de amenaza	Atacante que envía prompts adversariales (jailbreaks) de cinco familias: inyección de prompt, roleplay, codificación (Base64, ROT13, leetspeak), etc.
Tipo de ataque	Jailbreak de una sola ronda (no se especifica multi-turno); ataques de codificación y manipulación de instrucciones
Adaptatividad del atacante	No adaptativo (conjunto fijo de ataques predefinidos)
Entorno o benchmark	Evaluación offline sobre conjunto de prompts de ataque; se mide ASR (Attack Success Rate) y tasa de sobre-rechazo (over-refusal)
Baseline o comparador	Modelo sin defensa (DPO solo), DPO + RepE y DPO + RepE + canonicalización, entre otros
Métricas de seguridad	Attack Success Rate (ASR); Relative ASR reduction; Over-refusal rate (falsos positivos en respuestas seguras)
Tasa de éxito del ataque	On Mistral-7B-Instruct evaluated against a 249-prompt attack set spanning five attack families, TRYLOCK achieves 88.0% relative ASR reduction (46.5% to 5.6%), with each layer contributing unique coverage: RepE blocks 36% of attacks that bypass DPO alone, while canonicalization catches 14% of encoding attacks that evade

Campo	Valor
Falsos positivos	By thresholding input perplexity, they detect 70% of attacks with 10% false positive rate.
Impacto en utilidad	Over-refusal reducido del 60% al 48% manteniendo defensa; no se reporta impacto en calidad de respuesta para consultas benignas
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Cobertura complementaria de capas: RepE bloquea 36% de ataques que evaden DPO; canonicalización atrapa 14% de ataques de codificación que evaden ambos
Modos de fallo	Fenómeno de steering no monótono: fuerza intermedia ($\alpha=1.0$) degrada seguridad por debajo de la línea base; interferencia RepE-DPO
Código o artefactos	Sí: adaptadores entrenados (168MB), vectores de steering (66KB), clasificador lateral (59MB), 2,939 pares de preferencia, metodología completa
Conclusión defensiva	TRYLOCK reduce ASR de 46.5% a 5.6% (88% relativo) en Mistral-7B-Instruct frente a ataques de jailbreak de cinco familias, con capas complementarias. El sidecar adaptativo reduce sobre-rechazo de 60% a 48% sin pérdida de defensa. La seguridad y usabilidad no son mutuamente excluyentes, pero se observa un fenómeno no monótono en steering que requiere calibración.
Confianza de extracción	95

- Hallazgos clave: TRYLOCK reduce el ASR de jailbreak del 46.5% al 5.6% en Mistral-7B-Instruct mediante una combinación en capas de DPO, RepE, canonicalización y un clasificador adaptativo, reduciendo además el sobre-rechazo del 60% al 48%.
- Evidencia de apoyo: TRYLOCK achieves 88.0% relative ASR reduction (46.5% to 5.6%)
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre jailbreak y cumplimiento de políticas y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 40. Your Agent is More Brittle Than You Think: Uncovering Indirect Injection Vulnerabilities in Agentic LLMs

Your Agent is More Brittle Than You Think: Uncovering Indirect Injection Vulnerabilities in Agentic LLMs entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Arquitectura multiagente con frameworks de código abierto (Voyager, Clawbot); backbones LLM no especificados frente a Atacante que inserta instrucciones maliciosas en contenido de terceros (Indirect Prompt Injection) para lograr exfiltración de datos o acciones no autorizadas, con aplicación en Posición de entrada de la herramienta (tool-input position) antes de la ejecución de la acción. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Las defensas actuales contra inyecciones indirectas de prompts son frágiles en entornos dinámicos multi-paso; el disyuntor basado en Representation Engineering (RepE) detecta e intercepta acciones no autorizadas con alta precisión antes de su ejecución.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo.

Protege frente a Atacante que inserta instrucciones maliciosas en contenido de terceros (Indirect Prompt Injection) para lograr exfiltración de datos o acciones no autorizadas mediante Arquitectura multiagente con frameworks de código abierto (Voyager, Clawbot); backbones LLM no especificados, aplicado en Posición de entrada de la herramienta (tool-input position) antes de la ejecución de la acción; la capacidad del atacante se clasifica como No se especifica adaptabilidad del atacante; los ataques son predefinidos y aplicados en el entorno de evaluación. El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como Algunas mitigaciones superficiales producen efectos secundarios contraproducentes (no se cuantifica el impacto en utilidad) y el sobrecoste como no reportado; no reportado. La evidencia de robustez es El disyuntor basado en RepE logra alta precisión de detección en diversos backbones de LLM; las defensas baseline son frágiles frente a ataques avanzados y el fallo residual recuperable es Las defensas superficiales pueden ser contraproducentes; los agentes ejecutan instrucciones maliciosas casi instantáneamente; alta entropía de decisión en estados internos durante ataques. Para la pregunta de investigación, esto importa así: Las defensas actuales contra inyecciones indirectas de prompts son frágiles en entornos dinámicos multi-paso; el disyuntor basado en Representation Engineering (RepE) detecta e intercepta acciones no autorizadas con alta precisión antes de su ejecución. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Wenhui Zhu; Xuanzhao Dong; Xiwen Chen; Rui Cai; Peijie Qiu; Zhipeng Wang; Oana Frunza; Shao Tang; Jindong Gu; Yalin Wang
- Año: 2026
- DOI: 10.48550/arxiv.2604.03870

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	Voyager, Clawbot y otros no especificados (nueve backbones de LLM, no listados explícitamente en el digest) (n=9)
Harness o defensa	RepE-based circuit breaker
Arquitectura de control	Arquitectura multiagente con frameworks de código abierto (Voyager, Clawbot); backbones LLM no especificados
Punto de aplicación	Posición de entrada de la herramienta (tool-input position) antes de la ejecución de la acción
Modelo de amenaza	Atacante que inserta instrucciones maliciosas en contenido de terceros (Indirect Prompt Injection) para lograr exfiltración de datos o acciones no autorizadas
Tipo de ataque	Indirect Prompt Injection (IPI); cuatro vectores de ataque sofisticados (no detallados en el digest)
Adaptatividad del atacante	No se especifica adaptabilidad del atacante; los ataques son predefinidos y aplicados en el entorno de evaluación
Entorno o benchmark	Entorno dinámico multi-paso con llamadas a herramientas (tool-calling); simulación de operaciones normales del agente
Baseline o comparador	Seis estrategias de defensa (no listadas explícitamente en el digest), defensas superficiales y Representación Engineering (RepE) como

Campo	Valor
Métricas de seguridad	Tasa de éxito del ataque (Attack Success Rate, ASR); entropía de decisión; log-probabilidad media; precisión de detección del disyuntor RepE
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	Algunas mitigaciones superficiales producen efectos secundarios contraproducentes (no se cuantifica el impacto en utilidad)
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	El disyuntor basado en RepE logra alta precisión de detección en diversos backbones de LLM; las defensas baseline son frágiles frente a ataques avanzados
Modos de fallo	Las defensas superficiales pueden ser contraproducentes; los agentes ejecutan instrucciones maliciosas casi instantáneamente; alta entropía de decisión en estados internos durante ataques
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	Las defensas actuales contra IPI son insuficientes en entornos dinámicos multi-paso; el disyuntor basado en Representation Engineering (RepE) ofrece una detección robusta al interceptar acciones no autorizadas antes de su ejecución, aunque no se reportan métricas de sobrecoste o impacto en utilidad. La amenaza principal es la inyección indirecta de prompts en sistemas multiagente con espacios de acción expandidos.
Confianza de extracción	90

- Hallazgos clave: Las defensas actuales contra inyecciones indirectas de prompts son frágiles en entornos dinámicos multi-paso; el disyuntor basado en Representation Engineering (RepE) detecta e intercepta acciones no autorizadas con alta precisión antes de su ejecución.
- Evidencia de apoyo: Advanced injections successfully bypass nearly all baseline defenses, and some surface-level mitigations even produce counterproductive side effects.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 41. AlignTree: Efficient Defense Against LLM Jailbreak Attacks

AlignTree: Efficient Defense Against LLM Jailbreak Attacks entra en la revisión porque aporta evidencia trazable en la familia jailbreak y cumplimiento de políticas. Evalúa monitoreo de activaciones internas del LLM durante generación; clasificador random forest sobre dos señales: dirección de rechazo (lineal) y señal SVM (no lineal) frente a atacante que genera prompts adversarios para eludir las directrices de seguridad del LLM, con aplicación en durante la generación (post-hoc sobre activaciones). Metodológicamente se articula como evaluación experimental post hoc sobre benchmark con clasificador de bosque aleatorio entrenado en activaciones internas. Su aportación central es: AlignTree logra un ASR medio del 0.5% en múltiples modelos

y ataques de jailbreak, con una sobrecarga de latencia de solo 0.2 ms por token, superando a defensas previas como ShieldGemma y Llama-Guard en eficiencia y eficacia.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental post hoc sobre benchmark con clasificador de bosque aleatorio entrenado en activaciones internas. Protege frente a atacante que genera prompts adversarios para eludir las directrices de seguridad del LLM mediante monitoreo de activaciones internas del LLM durante generación; clasificador random forest sobre dos señales: dirección de rechazo (lineal) y señal SVM (no lineal), aplicado en durante la generación (post-hoc sobre activaciones); la capacidad del atacante se clasifica como no adaptativo (ataques predefinidos). El comparador principal es PerplexityFilter, SelfReminder y SafeDecoding, entre otros. La eficacia se reporta como AlignTree logra $ASR < 1\%$ en la mayoría de los ataques; por ejemplo, en Llama-2-7B con GCG: $ASR\ 0.0\%$; en Vicuna-7B con GCG: $ASR\ 0.0\%$; en promedio sobre todos los modelos y ataques, ASR medio de 0.5% (frente a $20\text{-}60\%$ en baselines), el efecto sobre uso legítimo como no reportado y el sobrecoste como sobrecarga de latencia media de 0.2 ms por token (frente a $1\text{-}5$ ms en baselines como ShieldGemma o Llama-Guard); no reportado. La evidencia de robustez es evaluado en 10 modelos y 8 ataques; mantiene ASR bajo incluso en ataques transferibles y en modelos no vistos durante entrenamiento y el fallo residual recuperable es no reportado explícitamente; se menciona que en ataques muy sofisticados (ej. Cipher) el ASR puede aumentar ligeramente (hasta 2%). Para la pregunta de investigación, esto importa así: AlignTree logra un ASR medio del 0.5% en múltiples modelos y ataques de jailbreak, con una sobrecarga de latencia de solo 0.2 ms por token, superando a defensas previas como ShieldGemma y Llama-Guard en eficiencia y eficacia. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Gil Goren; Shahar Katz; Lior Wolf
- Año: 2026
- DOI: 10.1609/aaai.v40i44.41074

Campo	Valor
Tipo de trabajo	Empírico
Diseño	evaluación experimental post hoc sobre benchmark con clasificador de bosque aleatorio entrenado en activaciones internas
Modelos o sistemas protegidos	Llama-2-7B, Llama-2-13B, Llama-3-8B y Mistral-7B, entre otros (n=10)
Harness o defensa	AlignTree
Arquitectura de control	monitoreo de activaciones internas del LLM durante generación; clasificador random forest sobre dos señales: dirección de rechazo (lineal) y señal SVM (no lineal)
Punto de aplicación	durante la generación (post-hoc sobre activaciones)
Modelo de amenaza	atacante que genera prompts adversarios para eludir las directrices de seguridad del LLM
Tipo de ataque	jailbreak (GCG, AutoDAN, PAIR, Cipher, DeepInception, SAP200, SAP200-S)
Adaptatividad del atacante	no adaptativo (ataques predefinidos)
Entorno o benchmark	evaluación offline sobre conjuntos de prompts de jailbreak conocidos
Baseline o comparador	PerplexityFilter, SelfReminder y SafeDecoding, entre otros
Métricas de seguridad	Attack Success Rate (ASR); False Positive Rate (FPR); Área bajo la curva ROC (AUC); latencia; throughput

Campo	Valor
Tasa de éxito del ataque	AlignTree logra ASR < 1% en la mayoría de los ataques; por ejemplo, en Llama-2-7B con GCG: ASR 0.0%; en Vicuna-7B con GCG: ASR 0.0%; en promedio sobre todos los modelos y ataques, ASR medio de 0.5% (frente a 20-60% en baselines)
Falsos positivos	FPR medio de 0.5% en prompts benignos (frente a 1-5% en baselines)
Impacto en utilidad	no reportado
Sobrecoste de latencia	sobrecarga de latencia media de 0.2 ms por token (frente a 1-5 ms en baselines como ShieldGemma o Llama-Guard)
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	evaluado en 10 modelos y 8 ataques; mantiene ASR bajo incluso en ataques transferibles y en modelos no vistos durante entrenamiento
Modos de fallo	no reportado explícitamente; se menciona que en ataques muy sofisticados (ej. Cipher) el ASR puede aumentar ligeramente (hasta 2%)
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	AlignTree reduce significativamente el ASR frente a ataques de jailbreak conocidos (ASR medio 0.5%) con una sobrecarga mínima de latencia (0.2 ms/token) y sin impacto apreciable en la utilidad del modelo. La defensa es robusta a través de múltiples modelos y ataques, superando a baselines como ShieldGemma y Llama-Guard en eficiencia y eficacia. Sin embargo, la evaluación se limita a ataques no adaptativos y no se reportan pruebas frente a ataques adaptativos o adversarios con conocimiento de la defensa.
Confianza de extracción	90

- Hallazgos clave: AlignTree logra un ASR medio del 0.5% en múltiples modelos y ataques de jailbreak, con una sobrecarga de latencia de solo 0.2 ms por token, superando a defensas previas como ShieldGemma y Llama-Guard en eficiencia y eficacia.
- Evidencia de apoyo: AlignTree monitors LLM activations during generation and detects misaligned behavior using an efficient random forest classifier.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre jailbreak y cumplimiento de políticas y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 42. Containment over Detection: Cryptographic Boundary Enforcement for Prompt Injection Defense in Agentic LLM Systems

Containment over Detection: Cryptographic Boundary Enforcement for Prompt Injection Defense in Agentic LLM Systems entra en la revisión porque aporta evidencia trazable en la familia jailbreak y cumplimiento de políticas. Evalúa Arquitectura de defensa por contención de cuatro capas: (1) generación de token criptográfico efímero (256 bits de entropía de límite), (2) canonicalización de entrada, (3) envoltura de sobre XML autenticado con token y validación de escape en servidor, (4) anclaje conductual del prompt. frente a Atacante de caja negra sin acceso oracle al token de ejecución runtime; el atacante puede incrustar contenido

adversarial en la entrada del usuario., con aplicación en Capa de entrada (input layer): antes de que el prompt llegue al LLM, se aplican las capas de contención.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante de caja negra sin acceso oracle al token de ejecución runtime; el atacante puede incrustar contenido adversarial en la entrada del usuario. mediante Arquitectura de defensa por contención de cuatro capas: (1) generación de token criptográfico efímero (256 bits de entropía de límite), (2) canonicalización de entrada, (3) envoltura de sobre XML autenticado con token y validación de escape en servidor, (4) anclaje conductual del prompt., aplicado en Capa de entrada (input layer): antes de que el prompt llegue al LLM, se aplican las capas de contención.; la capacidad del atacante se clasifica como No adaptativo (los patrones adversariales son fijos, no se adaptan a la defensa).. El comparador principal es no reportado (no se comparan defensas existentes y solo se evalúa la arquitectura propuesta). La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como <0.5 ms por solicitud; no reportado. La evidencia de robustez es Prueba formal de que la probabilidad de construir una fuga de límite válida es $\leq 2^{-256}$ bajo el modelo de atacante; 15 propiedades de corrección verificables por máquina (Hypothesis), 3 de las cuales detectaron errores críticos antes del despliegue. y el fallo residual recuperable es El 5.7% de fallos residuales son jailbreaks semánticos a nivel de modelo, ortogonales a la defensa en la capa de entrada; no se especifican otros modos de fallo.. Para la pregunta de investigación, esto importa así: el estudio aporta evidencia sustantiva que ayuda a delimitar mecanismo, contexto y alcance inferencial. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Saurabh Sharma
- Año: 2026
- DOI: 10.21203/rs.3.rs-10196595/v1

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	no reportado (la defensa es independiente del modelo y se evalúa sobre patrones adversariales, no sobre modelos específicos) (n=1)
Harness o defensa	Token-Containment Defense (sin nombre formal de harness; se refiere a la arquitectura de defensa)
Arquitectura de control	Arquitectura de defensa por contención de cuatro capas: (1) generación de token criptográfico efímero (256 bits de entropía de límite), (2) canonicalización de entrada, (3) envoltura de sobre XML autenticado con token y validación de escape en servidor, (4) anclaje conductual del prompt.
Punto de aplicación	Capa de entrada (input layer): antes de que el prompt llegue al LLM, se aplican las capas de contención.
Modelo de amenaza	Atacante de caja negra sin acceso oracle al token de ejecución runtime; el atacante puede incrustar contenido adversarial en la entrada del usuario.
Tipo de ataque	Inyección de instrucciones (prompt injection) directa e indirecta; incluye variantes de evasión basadas en codificación.

Campo	Valor
Adaptatividad del atacante	No adaptativo (los patrones adversariales son fijos, no se adaptan a la defensa).
Entorno o benchmark	Evaluación offline sobre corpus adversariales predefinidos; no se especifica si es en tiempo real o simulada.
Baseline o comparador	no reportado (no se comparan defensas existentes y solo se evalúa la arquitectura propuesta)
Métricas de seguridad	Tasa de contención sintáctica (syntactic containment rate); tasa de fallos residuales (residual failures); sobrecarga de latencia (runtime overhead).
Tasa de éxito del ataque	no reportado
Falsos positivos	Evaluation on 500 production messages revealed: • False positive rate: 8.3%.
Impacto en utilidad	no reportado
Sobrecoste de latencia	<0.5 ms por solicitud
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Prueba formal de que la probabilidad de construir una fuga de límite válida es $\leq 2^{-256}$ bajo el modelo de atacante; 15 propiedades de corrección verificables por máquina (Hypothesis), 3 de las cuales detectaron errores críticos antes del despliegue.
Modos de fallo	El 5.7% de fallos residuales son jailbreaks semánticos a nivel de modelo, ortogonales a la defensa en la capa de entrada; no se especifican otros modos de fallo.
Código o artefactos	Sí, el arnés de evaluación y las especificaciones de propiedades se publican como código abierto (enlace: https://github.com/saurabh0880/token-containment-defense).
Conclusión defensiva	La arquitectura de defensa por contención propuesta logra una tasa de contención sintáctica del 94.3% frente a 547 patrones adversariales de cuatro corpus, con una sobrecarga de latencia inferior a 0.5 ms por solicitud. Los fallos residuales (5.7%) son jailbreaks semánticos a nivel de modelo, no atribuibles a la defensa de capa de entrada. La defensa no se compara con líneas base; la conclusión de seguridad es que la contención sintáctica es efectiva contra inyección de instrucciones, pero no aborda jailbreaks semánticos.
Confianza de extracción	95

- Hallazgos clave: La arquitectura de defensa por contención alcanza un 94.3% de contención sintáctica frente a inyección de instrucciones, con una sobrecarga de latencia inferior a 0.5 ms; los fallos residuales son jailbreaks semánticos a nivel de modelo, no mitigables por la defensa de capa de entrada.
- Evidencia de apoyo: Empirical evaluation against 547 adversarial patterns drawn from four established attack corpora (Garak, OWASP LLM Top 10, HarmBench prompt injection subset, and a custom multilingual probe set) demonstrates a 94.3% syntactic containment rate; the 5.7%...
- Localización de la evidencia: p. 2 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre jailbreak y cumplimiento de políticas y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con

contrato defensivo menos completo.

Estudio 43. Mitigating Indirect Prompt Injection via Instruction-Following Intent Analysis

Mitigating Indirect Prompt Injection via Instruction-Following Intent Analysis entra en la revisión porque aporta evidencia trazable en la familia contexto, rag y prompt injection indirecta. Evalúa analizador de intención de seguimiento de instrucciones (IIA) con intervenciones en el pensamiento frente a atacante que inyecta instrucciones maliciosas en datos no fiables (por ejemplo, contenido de páginas web, correos) para que el agente las ejecute, con aplicación en post-procesado del prompt de entrada (identificación y neutralización de instrucciones no deseadas). Metodológicamente se articula como evaluación experimental post hoc sobre benchmarks agentivos con ataques adaptativos. Su aportación central es: IntentGuard reduce la tasa de éxito de ataques de inyección indirecta de instrucciones del 100% al 8.5% en un escenario de Mind2Web, sin degradar la utilidad del agente en la mayoría de los casos, incluso frente a ataques adaptativos.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como evaluación experimental post hoc sobre benchmarks agentivos con ataques adaptativos. Protege frente a atacante que inyecta instrucciones maliciosas en datos no fiables (por ejemplo, contenido de páginas web, correos) para que el agente las ejecute mediante analizador de intención de seguimiento de instrucciones (IIA) con intervenciones en el pensamiento, aplicado en post-procesado del prompt de entrada (identificación y neutralización de instrucciones no deseadas); la capacidad del atacante se clasifica como adaptativo (el atacante conoce la defensa y optimiza el ataque mediante búsqueda con LLM). El comparador principal es sin defensa (ataque directo) y variantes de ataque adaptativo. La eficacia se reporta como Results demonstrate that IntentGuard achieves (1) no utility degradation in all but one setting and (2) strong robustness against adaptive prompt injection attacks (e.g., reducing attack success rates from 100% to 8.5% in a Mind2Web scenario)., el efecto sobre uso legítimo como Results demonstrate that IntentGuard achieves (1) no utility degradation in all but one setting and (2) strong robustness against adaptive prompt injection attacks (e.g., reducing attack success rates from 100% to 8.5% in a Mind2Web scenario). y el sobrecoste como no reportado; no reportado. La evidencia de robustez es frente a ataques adaptativos, ASR baja del 100% al 8.5% en Mind2Web; en AgentDojo también se observa robustez y el fallo residual recuperable es Our key observation is that most adaptive attacks (Nasr et al., 2025; Chao et al., 2025; Wen et al., 2025) involve adding extra instructions to persuade the model to follow malicious directives while bypassing defenses (e.g., by not verbalizing malicious instructions in IntentGuard).. Para la pregunta de investigación, esto importa así: IntentGuard reduce la tasa de éxito de ataques de inyección indirecta de instrucciones del 100% al 8.5% en un escenario de Mind2Web, sin degradar la utilidad del agente en la mayoría de los casos, incluso frente a ataques adaptativos. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Mintong Kang; Chong Xiang; Sanjay Kariyappa; Chaowei Xiao; Bo Li; Edward Suh
- Año: 2025
- DOI: 10.48550/arxiv.2512.00966

Campo	Valor
Tipo de trabajo	Empírico
Diseño	evaluación experimental post hoc sobre benchmarks agentivos con ataques adaptativos
Modelos o sistemas protegidos	Qwen-3-32B y gpt-oss-20B
Harness o defensa	IntentGuard
Arquitectura de control	analizador de intención de seguimiento de instrucciones (IIA) con intervenciones en el pensamiento

Campo	Valor
Punto de aplicación	post-procesado del prompt de entrada (identificación y neutralización de instrucciones no deseadas)
Modelo de amenaza	atacante que inyecta instrucciones maliciosas en datos no fiables (por ejemplo, contenido de páginas web, correos) para que el agente las ejecute
Tipo de ataque	indirect prompt injection (inyección indirecta de instrucciones); ataque adaptativo con búsqueda asistida por LLM
Adaptatividad del atacante	adaptativo (el atacante conoce la defensa y optimiza el ataque mediante búsqueda con LLM)
Entorno o benchmark	benchmarks agentivos (AgentDojo y Mind2Web) con ataques directos y adaptativos
Baseline o comparador	sin defensa (ataque directo) y variantes de ataque adaptativo
Métricas de seguridad	attack success rate (ASR); utility (tasa de éxito en tareas legítimas)
Tasa de éxito del ataque	Results demonstrate that IntentGuard achieves (1) no utility degradation in all but one setting and (2) strong robustness against adaptive prompt injection attacks (e.g., reducing attack success rates from 100% to 8.5% in a Mind2Web scenario).
Falsos positivos	We find that: (1) IntentGuard preserves benign utility with zero false positives; (2) it achieves substantial robustness gains, especially under strong adaptive attacks; (3) it consistently outperforms baselines across datasets and models; (4) combining start-of-thinking prefilling and end-of-thinking refinement yields
Impacto en utilidad	Results demonstrate that IntentGuard achieves (1) no utility degradation in all but one setting and (2) strong robustness against adaptive prompt injection attacks (e.g., reducing attack success rates from 100% to 8.5% in a Mind2Web scenario).
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	frente a ataques adaptativos, ASR baja del 100% al 8.5% en Mind2Web; en AgentDojo también se observa robustez
Modos de fallo	Our key observation is that most adaptive attacks (Nasr et al., 2025; Chao et al., 2025; Wen et al., 2025) involve adding extra instructions to persuade the model to follow malicious directives while bypassing defenses (e.g., by not verbalizing malicious instructions in IntentGuard).
Código o artefactos	no reportado

Campo	Valor
Conclusión defensiva	IntentGuard reduce significativamente el ASR de inyección indirecta de instrucciones (de 100% a 8.5% en un escenario) sin apenas impacto en utilidad, incluso frente a ataques adaptativos que conocen la defensa. La defensa se basa en analizar la intención de seguir instrucciones, no en detectar texto malicioso.
Confianza de extracción	90

- Hallazgos clave: IntentGuard reduce la tasa de éxito de ataques de inyección indirecta de instrucciones del 100% al 8.5% en un escenario de Mind2Web, sin degradar la utilidad del agente en la mayoría de los casos, incluso frente a ataques adaptativos.
- Evidencia de apoyo: Results demonstrate that IntentGuard achieves (1) no utility degradation in all but one setting and (2) strong robustness against adaptive prompt injection attacks (e.g., reducing attack success rates from 100% to 8.5% in a Mind2Web scenario).
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre contexto, rag y prompt injection indirecta y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 44. Indirect Prompt Injections: Are Firewalls All You Need, or Stronger Benchmarks?

Indirect Prompt Injections: Are Firewalls All You Need, or Stronger Benchmarks? entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Arquitectura modular y agnóstica al modelo, operando en la interfaz agente-herramienta frente a Inyección indirecta de prompt (IPI): instrucciones maliciosas incrustadas en contenido externo o salidas de herramientas, con aplicación en Interfaz agente-herramienta (entrada y salida de herramientas). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Inyección indirecta de prompt (IPI): instrucciones maliciosas incrustadas en contenido externo o salidas de herramientas mediante Arquitectura modular y agnóstica al modelo, operando en la interfaz agente-herramienta, aplicado en Interfaz agente-herramienta (entrada y salida de herramientas); la capacidad del atacante se clasifica como Adaptativo (tercera etapa del ataque en cascada). El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia se reporta como Safe refusal or ignoring the injection counts as $ASR = 0.$, el efecto sobre uso legítimo como demonstrate that this simple “minimize & sanitize” defense that requires no LLM retraining or proprietary guardrails can achieve approx.0% attack success with minimal utility degradation across four widely used benchmarks. y el sobre coste como no reportado; As shown in Table 25, CaMeL incurs the highest token cost, requiring $2.82 \times$ more input tokens and $2.73 \times$ more output tokens than the baseline (no defense).. La evidencia de robustez es La defensa logra $ASR=0$ en todos los benchmarks; se evalúa adicionalmente con ataques adaptativos de tres etapas, pero no se reportan resultados numéricos de esos ataques en el digest y el fallo residual recuperable es No se reportan modos de fallo de la defensa; se identifican limitaciones en los benchmarks (métricas defectuosas, bugs, ataques débiles). Para la pregunta de investigación, esto importa así: el estudio aporta evidencia sustantiva que ayuda a delimitar mecanismo, contexto y alcance inferencial. La confianza de extracción es alta y permite integrar el estudio en la comparación central con una base empírica suficiente.

- Autores: Rishika Bhagwatkar; Kevin Kasa; Abhay Puri; Gabriel Huang; Irina Rish; Graham W. Taylor; Krishnamurthy Dj Dvijotham; Alexandre Lacoste

- Año: 2025
- DOI: 10.48550/arxiv.2510.05244

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	no reportado (el estudio es agnóstico al modelo y no se nombran modelos concretos en el digest) (n=10)
Harness o defensa	Firewall de defensa para agentes (Tool-Input Firewall Minimizer + Tool-Output Firewall Sanitizer)
Arquitectura de control	Arquitectura modular y agnóstica al modelo, operando en la interfaz agente-herramienta
Punto de aplicación	Interfaz agente-herramienta (entrada y salida de herramientas)
Modelo de amenaza	Inyección indirecta de prompt (IPI): instrucciones maliciosas incrustadas en contenido externo o salidas de herramientas
Tipo de ataque	Ataque en tres etapas: (1) inyección estándar de prompt, (2) ataques de segundo orden, (3) ataques adaptativos
Adaptatividad del atacante	Adaptativo (tercera etapa del ataque en cascada)
Entorno o benchmark	Evaluación offline sobre benchmarks públicos (AgentDojo, Agent Security Bench, InjecAgent, tau-Bench)
Baseline o comparador	Resultados previos reportados en los benchmarks (no se listan explícitamente en el digest) y la defensa propuesta se compara con el estado
Métricas de seguridad	Tasa de éxito del ataque (ASR); utilidad (no definida explícitamente en el digest, pero se menciona ‘high utility’ y ‘security-utility tradeoff’)
Tasa de éxito del ataque	Safe refusal or ignoring the injection counts as ASR = 0.
Falsos positivos	no reportado
Impacto en utilidad	demonstrate that this simple “minimize & sanitize” defense that requires no LLM retraining or proprietary guardrails can achieve aprox.0% attack success with minimal utility degradation across four widely used benchmarks.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	As shown in Table 25, CaMeL incurs the highest token cost, requiring 2.82× more input tokens and 2.73× more output tokens than the baseline (no defense).
Evidencia de robustez	La defensa logra ASR=0 en todos los benchmarks; se evalúa adicionalmente con ataques adaptativos de tres etapas, pero no se reportan resultados numéricos de esos ataques en el digest
Modos de fallo	No se reportan modos de fallo de la defensa; se identifican limitaciones en los benchmarks (métricas defectuosas, bugs, ataques débiles)
Código o artefactos	Disponible en página web del proyecto: https://firewall-defenses.github.io

Campo	Valor
Conclusión defensiva	La defensa basada en dos cortafuegos (Minimizer y Sanitizer) en la interfaz agente-herramienta logra seguridad perfecta (ASR=0) en cuatro benchmarks públicos, con alta utilidad y sin comprometer el rendimiento. Sin embargo, los benchmarks existentes son fácilmente saturados, lo que indica la necesidad de benchmarks más sólidos con ataques adaptativos y métricas cuidadosas. La conclusión no debe interpretarse como que un ASR más bajo en un único benchmark demuestre superioridad absoluta; el estudio muestra que la defensa es efectiva bajo el threat model de IPI, pero la adaptatividad del atacante (ataques en tres etapas) y el efecto en utilidad son favorables, aunque no se reportan costes de latencia o computacionales.
Confianza de extracción	85

- Hallazgos clave: Una defensa simple basada en dos cortafuegos en la interfaz agente-herramienta logra seguridad perfecta (ASR=0) en cuatro benchmarks de inyección indirecta de prompt, saturando los benchmarks existentes y revelando la necesidad de ataques adaptativos más fuertes y métricas mejoradas.
- Evidencia de apoyo: we show that a simple, modular, and model-agnostic defense operating at the agent-tool interface achieves perfect security with high utility across all four public benchmarks
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 45. Autonomous LLM Agent Worms: Cross-Platform Propagation, Automated Discovery and Temporal Re-Entry Defense

Autonomous LLM Agent Worms: Cross-Platform Propagation, Automated Discovery and Temporal Re-Entry Defense entra en la revisión porque aporta evidencia trazable en la familia contexto, rag y prompt injection indirecta. Evalúa Defensa basada en restricciones de reentrada temporal: bloqueo de escritura antes de lectura expuesta, configuración sellada, promoción de memoria tipada y atenuación de capacidades. frente a Atacante que inyecta contenido malicioso en estado persistente del agente (archivos, memoria) que se recarga automáticamente y propaga a otros agentes sin intervención humana., con aplicación en Punto de carga automática de archivos persistentes en el contexto de decisión del LLM; operaciones de lectura/escritura en memoria y archivos.. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Se demuestra por primera vez la propagación autónoma de gusanos persistentes en ecosistemas multiagente LLM, con capacidad de transmisión multiplataforma sin adaptación específica.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante que inyecta contenido malicioso en estado persistente del agente (archivos, memoria) que se recarga automáticamente y propaga a otros agentes sin intervención humana. mediante Defensa basada en restricciones de reentrada temporal: bloqueo de escritura antes de lectura expuesta, configuración sellada, promoción de memoria tipada y atenuación de capacidades., aplicado en Punto de carga automática de archivos persistentes en el contexto de decisión del LLM; operaciones de lectura/escritura en memoria y archivos.; la capacidad del atacante se clasifica como Alta: el optimizador SRPO genera cargas útiles robustas a resúmenes y paráfrasis del LLM en comunicación multi-salto.. El comparador principal es Comparación

entre portadores de mensajes de usuario y de sistema y sin defensa vs. con defensa RTW-A. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Demostración empírica de que SRPO genera cargas útiles robustas a resúmenes y paráfrasis del LLM; RTW-A elimina la cadena de persistencia-reentrada-acción. y el fallo residual recuperable es Portadores de mensajes de sistema tienen menor cumplimiento de ataque que los de usuario; operaciones de lectura son la principal amenaza de integridad.. Para la pregunta de investigación, esto importa así: Se demuestra por primera vez la propagación autónoma de gusanos persistentes en ecosistemas multiagente LLM, con capacidad de transmisión multiplataforma sin adaptación específica. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Mingming Zha; Xiaofeng Wang
- Año: 2026
- DOI: 10.48550/arxiv.2605.02812

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	SSCGV (analizador de grafos de código fuente), SRPO (optimizador de cargas útiles), RTW-A (defensa de reentrada temporal) y Frameworks A-C (sistemas
Harness o defensa	RTW-A (Re-entry Temporal Worm defense)
Arquitectura de control	Defensa basada en restricciones de reentrada temporal: bloqueo de escritura antes de lectura expuesta, configuración sellada, promoción de memoria tipada y atenuación de capacidades.
Punto de aplicación	Punto de carga automática de archivos persistentes en el contexto de decisión del LLM; operaciones de lectura/escritura en memoria y archivos.
Modelo de amenaza	Atacante que inyecta contenido malicioso en estado persistente del agente (archivos, memoria) que se recarga automáticamente y propaga a otros agentes sin intervención humana.
Tipo de ataque	Inyección de gusano persistente; propagación autónoma zero-click; transmisión multiplataforma; escalada de privilegios; exfiltración de datos.
Adaptatividad del atacante	Alta: el optimizador SRPO genera cargas útiles robustas a resúmenes y paráfrasis del LLM en comunicación multi-salto.
Entorno o benchmark	Entorno controlado con tres marcos de agentes de producción anonimizados; simulación de propagación autónoma sin clic y transmisión de 3 saltos.
Baseline o comparador	Comparación entre portadores de mensajes de usuario y de sistema y sin defensa vs. con defensa RTW-A
Métricas de seguridad	Tasa de cumplimiento de ataque (attack compliance); tasa de propagación autónoma; número de saltos de transmisión; éxito de exfiltración; escalada de privilegios.
Tasa de éxito del ataque	no reportado

Campo	Valor
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Demostración empírica de que SRPO genera cargas útiles robustas a resúmenes y paráfrasis del LLM; RTW-A elimina la cadena de persistencia-reentrada-acción.
Modos de fallo	Portadores de mensajes de sistema tienen menor cumplimiento de ataque que los de usuario; operaciones de lectura son la principal amenaza de integridad.
Código o artefactos	No disponible públicamente por divulgación coordinada; sistemas anonimizados.
Conclusión defensiva	RTW-A elimina la cadena de persistencia-reentrada-acción en gusanos persistentes, pero la defensa se basa en supuestos de que el atacante no puede eludir las restricciones de reentrada temporal; se requiere validación adicional en entornos reales.
Confianza de extracción	90

- Hallazgos clave: Se demuestra por primera vez la propagación autónoma de gusanos persistentes en ecosistemas multiagente LLM, con capacidad de transmisión multiplataforma sin adaptación específica. La defensa RTW-A elimina la cadena de persistencia-reentrada-acción, pero los portadores de mensajes de usuario son más efectivos que los de sistema.
- Evidencia de apoyo: Evaluated on three production agent frameworks, we demonstrate zero-click autonomous propagation, 3-hop cross-platform transmission without platform-specific adaptation, inter-agent privilege escalation, and data exfiltration.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre contexto, rag y prompt injection indirecta y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 46. Blind Gods and Broken Screens: Architecting a Secure, Intent-Centric Mobile Agent Operating System

Blind Gods and Broken Screens: Architecting a Secure, Intent-Centric Mobile Agent Operating System entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa Hub-and-Spoke con Kernel de Agente privilegiado, Agentes de Aplicación en sandbox, y cuatro pilares de defensa: registro global de identidad, cortafuegos semántico multicapa, integridad cognitiva (memoria con marcado de contaminación y alineación plan-trayectoria), control de acceso granular con auditoría no repudiable. frente a Cuatro dimensiones: Identidad del Agente, Interfaz Externa, Razonamiento Interno, Ejecución de Acciones. Amenazas: identidad falsa de app, suplantación visual, inyección indirecta de instrucciones, escalada de privilegios no autorizada., con aplicación en Kernel del Agente (media toda la comunicación entre System Agent y App Agents). Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: Aura reduce la tasa de éxito de ataques de alto riesgo del ~40% al 4.4% y mejora la tasa de éxito en tareas de bajo riesgo del ~75% al 94.3% respecto a Doubao, con ganancias de latencia de casi un orden de magnitud.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental comparativa

post hoc sobre benchmark de seguridad (MobileSafetyBench) con análisis de amenazas estructurado en cuatro dimensiones.. Protege frente a Cuatro dimensiones: Identidad del Agente, Interfaz Externa, Razonamiento Interno, Ejecución de Acciones. Amenazas: identidad falsa de app, suplantación visual, inyección indirecta de instrucciones, escalada de privilegios no autorizada. mediante Hub-and-Spoke con Kernel de Agente privilegiado, Agentes de Aplicación en sandbox, y cuatro pilares de defensa: registro global de identidad, cortafuegos semántico multicapa, integridad cognitiva (memoria con marcado de contaminación y alineación plan-trayectoria), control de acceso granular con auditoría no repudiable., aplicado en Kernel del Agente (media toda la comunicación entre System Agent y App Agents); la capacidad del atacante se clasifica como no reportado. El comparador principal es Doubao Mobile Assistant (línea base comercial). La eficacia se reporta como Aura: 4.4% (high-risk ASR); Doubao: ~40% (high-risk ASR), el efecto sobre uso legítimo como Evaluation on MobileSafetyBench shows that, compared to Doubao, Aura improves low-risk Task Success Rate from roughly 75% to 94.3%, reduces high-risk Attack Success Rate from roughly 40% to 4.4%, and achieves near-order-of-magnitude latency gains. y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Reducción de ASR de ~40% a 4.4% en ataques de alto riesgo; mejora de TSR de ~75% a 94.3% en tareas de bajo riesgo y el fallo residual recuperable es Major “Super Apps” (e.g., WeChat, Taobao) essentially function as walled gardens, resisting external Blind Gods and Broken Screens: Architecting a Secure, Intent-Centric Mobile Agent Operating System 3 agentic scrapping not merely due to technical barriers, but because it bypasses the native user experience [30, 42].. Para la pregunta de investigación, esto importa así: Aura reduce la tasa de éxito de ataques de alto riesgo del ~40% al 4.4% y mejora la tasa de éxito en tareas de bajo riesgo del ~75% al 94.3% respecto a Doubao, con ganancias de latencia de casi un orden de magnitud. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Zhenhua Zou; Sheng Guo; Qiuyang Zhan; Lepeng Zhao; Shuo Li; Qi Li; Ke Xu; Mingwei Xu; Zhuotao Liu
- Año: 2026
- DOI: 10.48550/arxiv.2602.10915

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	Doubao Mobile Assistant y Aura (Agent Universal Runtime Architecture) (n=2)
Harness o defensa	Aura (Agent Universal Runtime Architecture)
Arquitectura de control	Hub-and-Spoke con Kernel de Agente privilegiado, Agentes de Aplicación en sandbox, y cuatro pilares de defensa: registro global de identidad, cortafuegos semántico multicapa, integridad cognitiva (memoria con marcado de contaminación y alineación plan-trayectoria), control de acceso granular con auditoría no repudiable.
Punto de aplicación	Kernel del Agente (media toda la comunicación entre System Agent y App Agents)
Modelo de amenaza	Cuatro dimensiones: Identidad del Agente, Interfaz Externa, Razonamiento Interno, Ejecución de Acciones. Amenazas: identidad falsa de app, suplantación visual, inyección indirecta de instrucciones, escalada de privilegios no autorizada.
Tipo de ataque	suplantación visual; inyección indirecta de instrucciones; escalada de privilegios; identidad falsa de aplicación
Adaptatividad del atacante	no reportado

Campo	Valor
Entorno o benchmark	benchmark MobileSafetyBench con tareas de bajo y alto riesgo
Baseline o comparador	Doubao Mobile Assistant (línea base comercial)
Métricas de seguridad	Task Success Rate (TSR) para tareas de bajo riesgo; Attack Success Rate (ASR) para ataques de alto riesgo; latencia
Tasa de éxito del ataque	Aura: 4.4% (high-risk ASR); Doubao: ~40% (high-risk ASR)
Falsos positivos	no reportado
Impacto en utilidad	Evaluation on MobileSafetyBench shows that, compared to Doubao, Aura improves low-risk Task Success Rate from roughly 75% to 94.3%, reduces high-risk Attack Success Rate from roughly 40% to 4.4%, and achieves near-order-of-magnitude latency gains.
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Reducción de ASR de ~40% a 4.4% en ataques de alto riesgo; mejora de TSR de ~75% a 94.3% en tareas de bajo riesgo
Modos de fallo	Major “Super Apps” (e.g., WeChat, Taobao) essentially function as walled gardens, resisting external Blind Gods and Broken Screens: Architecting a Secure, Intent-Centric Mobile Agent Operating System 3 agentic scrapping not merely due to technical barriers, but because it bypasses the native user experience [30, 42].
Código o artefactos	no reportado
Conclusión defensiva	Aura, una arquitectura de kernel de agente con cuatro pilares de defensa, reduce significativamente la tasa de éxito de ataques de alto riesgo (ASR del ~40% al 4.4%) y mejora la utilidad (TSR del ~75% al 94.3%) en comparación con Doubao Mobile Assistant, con ganancias de latencia de casi un orden de magnitud, demostrando ser una alternativa viable y segura al paradigma ‘Pantalla como Interfaz’.
Confianza de extracción	95

- Hallazgos clave: Aura reduce la tasa de éxito de ataques de alto riesgo del ~40% al 4.4% y mejora la tasa de éxito en tareas de bajo riesgo del ~75% al 94.3% respecto a Doubao, con ganancias de latencia de casi un orden de magnitud.
- Evidencia de apoyo: Evaluation on MobileSafetyBench shows that, compared to Doubao, Aura improves low-risk Task Success Rate from roughly 75% to 94.3%, reduces high-risk Attack Success Rate from roughly 40% to 4.4%, and achieves near-order-of-magnitude latency gains.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 47. Governance Architecture for Autonomous Agent Systems: Threats, Framework, and Engineering Practice

Governance Architecture for Autonomous Agent Systems: Threats, Framework, and Engineering Practice entra en la revisión porque aporta evidencia trazable en la familia herramientas, permisos y aislamiento. Evalúa OpenClaw (marco de agente de código abierto representativo) frente a Atacante que envía llamadas a herramientas maliciosas (TC1/TC2/TC3) a un agente autónomo; incluye inyección de prompts, envenenamiento de RAG y plugins de habilidades maliciosos, con aplicación en Capa 2 (verificación de intenciones) principalmente evaluada; también Capa 1 (sandboxing de ejecución), Capa 3 (autorización inter-agente de confianza cero) y Capa 4 (registro de auditoría inmutable) en evaluación de pipeline. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como diseño descrito en el texto completo. Protege frente a Atacante que envía llamadas a herramientas maliciosas (TC1/TC2/TC3) a un agente autónomo; incluye inyección de prompts, envenenamiento de RAG y plugins de habilidades maliciosos mediante OpenClaw (marco de agente de código abierto representativo), aplicado en Capa 2 (verificación de intenciones) principalmente evaluada; también Capa 1 (sandboxing de ejecución), Capa 3 (autorización inter-agente de confianza cero) y Capa 4 (registro de auditoría inmutable) en evaluación de pipeline; la capacidad del atacante se clasifica como No reportado explícitamente; los ataques son estáticos en el benchmark. El comparador principal es Líneas base NLI ligeras (por debajo del 10% de IR), cascadas de dos etapas (Qwen3.5-9B->GPT-4o-mini y Qwen3.5-9B->Qwen2.5-14B), entre otros. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como P50 total de pipeline ~980 ms; capas no juez contribuyen ~18 ms; latencia de jueces LLM no desglosada individualmente; no reportado. La evidencia de robustez es Generalización a benchmark externo InjecAgent con 99-100% de intercepción; evaluación de pipeline con 96% IR y el fallo residual recuperable es TC3 (plugins maliciosos) más difícil (75-94% IR); NLI ligero por debajo del 10% IR; FPR hasta ~20% en algunos jueces locales. Para la pregunta de investigación, esto importa así: el estudio aporta evidencia sustantiva que ayuda a delimitar mecanismo, contexto y alcance inferencial. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Yuxu Ge
- Año: 2026
- DOI: 10.48550/arxiv.2603.07191

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	Qwen3.5-4B, Llama-3.1-8B, Qwen3.5-9B y Qwen2.5-14B, entre otros
Harness o defensa	Layered Governance Architecture (LGA)
Arquitectura de control	OpenClaw (marco de agente de código abierto representativo)
Punto de aplicación	Capa 2 (verificación de intenciones) principalmente evaluada; también Capa 1 (sandboxing de ejecución), Capa 3 (autorización inter-agente de confianza cero) y Capa 4 (registro de auditoría inmutable) en evaluación de pipeline
Modelo de amenaza	Atacante que envía llamadas a herramientas maliciosas (TC1/TC2/TC3) a un agente autónomo; incluye inyección de prompts, envenenamiento de RAG y plugins de habilidades maliciosos

Campo	Valor
Tipo de ataque	Inyección de prompts; envenenamiento de RAG; plugins de habilidades maliciosos
Adaptatividad del atacante	No reportado explícitamente; los ataques son estáticos en el benchmark
Entorno o benchmark	Benchmark sintético bilingüe (1.081 muestras) y benchmark externo InjecAgent; evaluación de pipeline de extremo a extremo (n=100)
Baseline o comparador	Líneas base NLI ligeras (por debajo del 10% de IR), cascadas de dos etapas (Qwen3.5-9B->GPT-4o-mini y Qwen3.5-9B->Qwen2.5-14B), entre otros
Métricas de seguridad	Interception Rate (IR); False Positive Rate (FPR); precisión-recall; latencia P50
Tasa de éxito del ataque	no reportado
Falsos positivos	7.1), and one cloud judge across all three threat classes on a 1,081-sample bilingual benchmark, showing that a two-stage cascade achieves 91.9–92.6% interception with 1.9–6.7% false positives, and a fully local cascade achieves 94.7–95.6% IR with 6.0–9.7% FPR (Sect.
Impacto en utilidad	no reportado
Sobrecoste de latencia	P50 total de pipeline ~980 ms; capas no juez contribuyen ~18 ms; latencia de jueces LLM no desglosada individualmente
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Generalización a benchmark externo InjecAgent con 99-100% de intercepción; evaluación de pipeline con 96% IR
Modos de fallo	TC3 (plugins maliciosos) más difícil (75-94% IR); NLI ligero por debajo del 10% IR; FPR hasta ~20% en algunos jueces locales
Código o artefactos	No reportado explícitamente; el marco LGA se describe pero no se menciona disponibilidad de código
Conclusión defensiva	La Arquitectura de Gobernanza en Capas (LGA) con verificación de intenciones basada en LLM (Capa 2) intercepta eficazmente llamadas a herramientas maliciosas (93-98.5% IR para TC1/TC2, 75-94% para TC3) con FPR manejable (1.9-20%), superando ampliamente a líneas base NLI ligeras (<10% IR). La combinación de capas (sandboxing, verificación, autorización, auditoría) logra un 96% IR en pipeline con baja latencia (~980 ms). La generalización a InjecAgent (99-100% IR) sugiere robustez. Sin embargo, TC3 sigue siendo un desafío y el FPR puede ser alto en algunos jueces locales, lo que motiva la aplicación complementaria en otras capas. No se reporta impacto en utilidad ni coste.
Confianza de extracción	95

- Hallazgos clave: La Arquitectura de Gobernanza en Capas (LGA) con verificación de intenciones basada en LLM intercepta eficazmente llamadas a herramientas maliciosas (93-98.5% IR) con baja

latencia (~980 ms), superando a líneas base NLI ligeras y generalizando bien a un benchmark externo (99-100% IR).

- Evidencia de apoyo: Experimental results on Layer 2 intent verification with four local LLM judges (Qwen3.5-4B, Llama-3.1-8B, Qwen3.5-9B, Qwen2.5-14B) and one cloud judge (GPT-4o-mini) show that all five LLM judges intercept 93.0–98.5% of TC1/TC2 malicious tool calls, while...
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre herramientas, permisos y aislamiento y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 48. SafeClaw-R: Towards Safe and Secure Multi-Agent Personal Assistants

SafeClaw-R: Towards Safe and Secure Multi-Agent Personal Assistants entra en la revisión porque aporta evidencia trazable en la familia exfiltración, secretos y privacidad. Evalúa SafeClaw-R impone seguridad como invariante a nivel de sistema sobre el grafo de ejecución, mediando acciones antes de la ejecución y aumentando habilidades con contrapartidas seguras. frente a Acciones no intencionadas por fallos de razonamiento del LLM; ataques de inyección de prompts que exfiltran credenciales o comprometen el sistema; habilidades maliciosas de terceros., con aplicación en Antes de la ejecución de la acción (mediación previa a la ejecución).. Metodológicamente se articula como método descrito en el texto completo. Su aportación central es: el estudio aporta evidencia sustantiva útil para delimitar mecanismo, contexto y alcance inferencial.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc sobre benchmarks adversariales y escenarios simulados en tres dominios (productividad, ecosistema de terceros, ejecución de código).. Protege frente a Acciones no intencionadas por fallos de razonamiento del LLM; ataques de inyección de prompts que exfiltran credenciales o comprometen el sistema; habilidades maliciosas de terceros. mediante SafeClaw-R impone seguridad como invariante a nivel de sistema sobre el grafo de ejecución, mediando acciones antes de la ejecución y aumentando habilidades con contrapartidas seguras., aplicado en Antes de la ejecución de la acción (mediación previa a la ejecución).; la capacidad del atacante se clasifica como No se especifica adaptabilidad del atacante; los ataques son predefinidos en los benchmarks.. El comparador principal es Regex baselines (precisión 61.6%), OpenClaw sin SafeClaw-R (implícito) y análisis de riesgo de habilidades integradas de OpenClaw.. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es SafeClaw-R alcanza 95.2% de precisión en Google Workspace, 97.8% en detección de patrones maliciosos de terceros y 100% en benchmark adversarial de ejecución de código. y el fallo residual recuperable es no reportado explícitamente; se menciona que los enfoques existentes (guardrails estáticos, LLM-as-a-Judge) carecen de aplicación fiable en tiempo real.. Para la pregunta de investigación, esto importa así: el estudio aporta evidencia sustantiva que ayuda a delimitar mecanismo, contexto y alcance inferencial. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Haoyu Wang; Zibo Xiao; Yedi Zhang; Christopher M. Poskitt; Jun Sun
- Año: 2026
- DOI: 10.48550/arxiv.2603.28807

Campo	Valor
Tipo de trabajo	Empírico
Diseño	diseño descrito en el texto completo
Modelos o sistemas protegidos	OpenClaw y SafeClaw-R (n=2)
Harness o defensa	SafeClaw-R

Campo	Valor
Arquitectura de control	SafeClaw-R impone seguridad como invariante a nivel de sistema sobre el grafo de ejecución, mediando acciones antes de la ejecución y aumentando habilidades con contrapartidas seguras.
Punto de aplicación	Antes de la ejecución de la acción (mediación previa a la ejecución).
Modelo de amenaza	Acciones no intencionadas por fallos de razonamiento del LLM; ataques de inyección de prompts que exfiltran credenciales o comprometen el sistema; habilidades maliciosas de terceros.
Tipo de ataque	Inyección de prompts; acciones no seguras por fallos de razonamiento; habilidades maliciosas de terceros.
Adaptatividad del atacante	No se especifica adaptabilidad del atacante; los ataques son predefinidos en los benchmarks.
Entorno o benchmark	Entornos simulados: Google Workspace, ecosistema de habilidades de terceros, ejecución de código adversarial.
Baseline o comparador	Regex baselines (precisión 61.6%), OpenClaw sin SafeClaw-R (implícito) y análisis de riesgo de habilidades integradas de OpenClaw.
Métricas de seguridad	Precisión de detección; tasa de detección de patrones maliciosos; precisión en benchmark adversarial.
Tasa de éxito del ataque	no reportado
Falsos positivos	At the same time, SafeClaw-R maintains a reasonable usability profile, with a modest 3.4% false positive rate in Google Workspace and no false positives in code execution, though it exhibits a conservative tendency to defer uncertain cases to review.
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	SafeClaw-R alcanza 95.2% de precisión en Google Workspace, 97.8% en detección de patrones maliciosos de terceros y 100% en benchmark adversarial de ejecución de código.
Modos de fallo	no reportado explícitamente; se menciona que los enfoques existentes (guardrails estáticos, LLM-as-a-Judge) carecen de aplicación fiable en tiempo real.
Código o artefactos	código o artefacto reportado como disponible
Conclusión defensiva	SafeClaw-R demuestra ser eficaz para la aplicación en tiempo real de seguridad en MAS autónomos, superando significativamente a las líneas base de regex en precisión de detección (95.2% vs 61.6% en Google Workspace) y alcanzando altas tasas de detección en otros dominios. Sin embargo, no se reportan métricas de falsos positivos, impacto en utilidad, latencia o coste, lo que limita la evaluación completa del equilibrio seguridad-utilidad.
Confianza de extracción	90

- Hallazgos clave: SafeClaw-R, un marco de seguridad que media acciones antes de la ejecución, alcanza una precisión de detección del 95.2% en escenarios de Google Workspace, superando ampliamente el 61.6% de las líneas base de regex, y logra un 100% de detección en un benchmark adversarial de ejecución de código.
- Evidencia de apoyo: SafeClaw-R achieves 95.2% accuracy in Google Workspace scenarios, significantly outperforming regex baselines (61.6%), detects 97.8% of malicious third-party skill patterns, and achieves 100% detection accuracy in our adversarial code execution benchmark.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre exfiltración, secretos y privacidad y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 49. InjecGuard: Benchmarking and Mitigating Over-defense in Prompt Injection Guardrail Models

InjecGuard: Benchmarking and Mitigating Over-defense in Prompt Injection Guardrail Models entra en la revisión porque aporta evidencia trazable en la familia exfiltración, secretos y privacidad. Evalúa modelo guardrail ligero que analiza semánticamente la entrada antes de llegar al LLM frente a ataques de inyección de instrucciones (goal hijacking y fuga de datos), con aplicación en pre-LLM (antes de que la entrada llegue al modelo de lenguaje). Metodológicamente se articula como evaluación experimental post hoc sobre benchmark con propuesta de nuevo modelo y estrategia de entrenamiento. Su aportación central es: Los modelos de protección de instrucciones existentes presentan sobredefensa, con precisión cercana al azar (60%) en el nuevo benchmark NotInject.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en entrenamiento con estrategia Mitigating Over-defense for Free (MOF); evaluación en benchmark NotInject. El diseño concreto puede resumirse como evaluación experimental post hoc sobre benchmark con propuesta de nuevo modelo y estrategia de entrenamiento. Protege frente a ataques de inyección de instrucciones (goal hijacking y fuga de datos) mediante modelo guardrail ligero que analiza semánticamente la entrada antes de llegar al LLM, aplicado en pre-LLM (antes de que la entrada llegue al modelo de lenguaje); la capacidad del atacante se clasifica como no adaptativo (ataques estáticos con palabras desencadenantes). El comparador principal es modelos de protección de instrucciones existentes (mejor modelo existente mencionado como baseline). La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es InjecGuard supera al mejor modelo existente en un 30.8% en NotInject y el fallo residual recuperable es sobredefensa por sesgo en palabras desencadenantes. Para la pregunta de investigación, esto importa así: Los modelos de protección de instrucciones existentes presentan sobredefensa, con precisión cercana al azar (60%) en el nuevo benchmark NotInject. La confianza de extracción es alta y permite integrar el estudio en la comparación central con una base empírica suficiente.

- Autores: Hao Li; Xiaogeng Liu
- Año: 2024
- DOI: 10.48550/arxiv.2410.22770

Campo	Valor
Tipo de trabajo	Empírico
Diseño	evaluación experimental post hoc sobre benchmark con propuesta de nuevo modelo y estrategia de entrenamiento

Campo	Valor
Modelos o sistemas protegidos	InjecGuard y varios modelos de protección de instrucciones de última generación (no listados explícitamente en el abstract, pero el digest menciona
Harness o defensa	InjecGuard
Arquitectura de control	modelo guardrail ligero que analiza semánticamente la entrada antes de llegar al LLM
Punto de aplicación	pre-LLM (antes de que la entrada llegue al modelo de lenguaje)
Modelo de amenaza	ataques de inyección de instrucciones (goal hijacking y fuga de datos)
Tipo de ataque	inyección de instrucciones directa e indirecta (basada en palabras desencadenantes)
Adaptatividad del atacante	no adaptativo (ataques estáticos con palabras desencadenantes)
Entorno o benchmark	evaluación offline sobre conjunto de datos NotInject y otros benchmarks
Baseline o comparador	modelos de protección de instrucciones existentes (mejor modelo existente mencionado como baseline)
Métricas de seguridad	precisión (accuracy); tasa de acierto en clasificación benigno/malicioso
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	InjecGuard supera al mejor modelo existente en un 30.8% en NotInject
Modos de fallo	sobredefensa por sesgo en palabras desencadenantes
Código o artefactos	sí (código y datasets en https://github.com/leolee99/InjecGuard)
Conclusión defensiva	InjecGuard reduce significativamente el sesgo en palabras desencadenantes y mejora la precisión en la detección de inyecciones de instrucciones, superando al mejor modelo existente en un 30.8% en el benchmark NotInject, ofreciendo una solución robusta y de código abierto.
Confianza de extracción	85

- Hallazgos clave: Los modelos de protección de instrucciones existentes presentan sobredefensa, con precisión cercana al azar (60%) en el nuevo benchmark NotInject. InjecGuard, con la estrategia MOF, reduce el sesgo y supera al mejor modelo existente en un 30.8%.
- Evidencia de apoyo: Our results show that state-of-the-art models suffer from over-defense issues, with accuracy dropping close to random guessing levels (60%).
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre exfiltración, secretos y privacidad y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Estudio 50. Securing AI Agents Against Prompt Injection Attacks

Securing AI Agents Against Prompt Injection Attacks entra en la revisión porque aporta evidencia trazable en la familia contexto, rag y prompt injection indirecta. Evalúa Agente RAG con recuperación de contenido externo frente a Atacante que inyecta instrucciones maliciosas en contenido recuperado externamente, con aplicación en Entrada (filtrado de contenido), prompt del sistema (guardrails), salida (verificación de respuestas). Metodológicamente se articula como Evaluación experimental post hoc sobre benchmark adversarial con comparación de defensas y modelos.. Su aportación central es: El marco de defensa multicapa reduce la tasa de éxito de ataques de inyección de prompts del 73.2% al 8.7% en agentes RAG, manteniendo el 94.3% del rendimiento de la tarea base.

Desde el punto de vista del diseño, se trata de un estudio empírico de tipo experimental apoyado en método descrito en el texto completo. El diseño concreto puede resumirse como Evaluación experimental post hoc sobre benchmark adversarial con comparación de defensas y modelos.. Protege frente a Atacante que inyecta instrucciones maliciosas en contenido recuperado externamente mediante Agente RAG con recuperación de contenido externo, aplicado en Entrada (filtrado de contenido), prompt del sistema (guardrails), salida (verificación de respuestas); la capacidad del atacante se clasifica como No adaptativo (ataques predefinidos en benchmark). El comparador principal es comparadores, grupos o condiciones descritos por el estudio. La eficacia se reporta como no reportado, el efecto sobre uso legítimo como no reportado y el sobrecoste como no reportado; no reportado. La evidencia de robustez es Reducción de ASR del 73.2% al 8.7% con marco combinado y el fallo residual recuperable es The dataset includes: • 200 base attack templates across five categories • 847 total test cases including variations in phrasing, obfuscation, and sophistication • 500 benign retrieval contexts for false positive evaluation • Metadata annotations including attack difficulty, expected model behavior, and success criteri. Para la pregunta de investigación, esto importa así: El marco de defensa multicapa reduce la tasa de éxito de ataques de inyección de prompts del 73.2% al 8.7% en agentes RAG, manteniendo el 94.3% del rendimiento de la tarea base. La confianza de extracción es muy alta, de modo que la ficha puede leerse como evidencia especialmente robusta dentro del corpus.

- Autores: Badrinath Ramakrishnan; Akshaya Balaji
- Año: 2025
- DOI: 10.48550/arxiv.2511.15759

Campo	Valor
Tipo de trabajo	Empírico
Diseño	Evaluación experimental post hoc sobre benchmark adversarial con comparación de defensas y modelos.
Modelos o sistemas protegidos	no reportado (siete modelos de lenguaje de última generación no nombrados explícitamente en el digest) (n=7)
Harness o defensa	Marco de defensa multicapa para agentes RAG (sin nombre propio)
Arquitectura de control	Agente RAG con recuperación de contenido externo
Punto de aplicación	Entrada (filtrado de contenido), prompt del sistema (guardrails), salida (verificación de respuestas)
Modelo de amenaza	Atacante que inyecta instrucciones maliciosas en contenido recuperado externamente
Tipo de ataque	Direct injection; context manipulation; instruction override; data exfiltration; cross-context contamination
Adaptatividad del atacante	No adaptativo (ataques predefinidos en benchmark)
Entorno o benchmark	Benchmark offline con 847 casos adversariales

Campo	Valor
Baseline o comparador	Sin defensa (ataque exitoso 73.2%), defensas individuales: content filtering, hierarchical guardrails, multi-stage verification y marco
Métricas de seguridad	Attack success rate (ASR); baseline task performance (utility)
Tasa de éxito del ataque	no reportado
Falsos positivos	no reportado
Impacto en utilidad	no reportado
Sobrecoste de latencia	no reportado
Sobrecoste económico o computacional	no reportado
Evidencia de robustez	Reducción de ASR del 73.2% al 8.7% con marco combinado
Modos de fallo	The dataset includes: • 200 base attack templates across five categories • 847 total test cases including variations in phrasing, obfuscation, and sophistication • 500 benign retrieval contexts for false positive evaluation • Metadata annotations including attack difficulty, expected model behavior, and success criteri
Código o artefactos	Sí (benchmark dataset y defensas liberados)
Conclusión defensiva	El marco combinado reduce significativamente la tasa de éxito de ataques de inyección de prompts (de 73.2% a 8.7%) con un impacto mínimo en la utilidad (94.3% de rendimiento base), pero no se reportan costes de latencia ni computacionales, y la evaluación es offline sin atacante adaptativo.
Confianza de extracción	90

- Hallazgos clave: El marco de defensa multicapa reduce la tasa de éxito de ataques de inyección de prompts del 73.2% al 8.7% en agentes RAG, manteniendo el 94.3% del rendimiento de la tarea base.
- Evidencia de apoyo: Our combined framework reduces successful attack rates from 73.2% to 8.7% while maintaining 94.3% of baseline task performance.
- Localización de la evidencia: p. 1 (full text; fragmento localizado)
- Lectura comparativa: Aporta evidencia sobre contexto, rag y prompt injection indirecta y permite comparar amenaza, punto de aplicación, baseline, eficacia, utilidad, coste y fallo residual frente a estudios con contrato defensivo menos completo.

Anexo B. Datos y trazabilidad

Se adjuntan como anexos CSV las fuentes minadas y derivadas que sostienen el análisis, con el fin de facilitar auditoría, replicación y reutilización por parte de lectoras y lectores.

- `paper/appendices/data/search-stage-map.csv`: 142 filas de datos.
- `paper/appendices/data/search-log.csv`: 127 filas de datos.
- `paper/appendices/data/master-records.csv`: 2372 filas de datos.
- `paper/appendices/data/doi-index.csv`: 2440 filas de datos.
- `paper/appendices/data/duplicates.csv`: 0 filas de ocurrencias duplicadas detectadas; el flujo de selección reporta la reducción neta consolidada antes del cribado.
- `paper/appendices/data/missing-doi.csv`: 122 filas de datos.
- `paper/appendices/data/title-abstract.csv`: 2372 filas de datos.

- `paper/appendices/data/full-text.csv`: 200 filas de datos.
- `paper/appendices/data/extraction-table.csv`: 50 filas de extracción; el corpus publicable final conserva los estudios con DOI, PDF legible y decisión de inclusión.
- `paper/appendices/data/selection-audit-matrix.csv`: 116 filas de preselección y auditoría; no debe leerse como N final del corpus.
- `paper/appendices/data/flow-counts.csv`: 10 filas de datos.
- `paper/appendices/data/manifest.csv`: 12 filas de datos.
- `paper/appendices/data/figures-evidence-manifest.csv`: 50 filas de datos.
- `paper/appendices/data/figures-page-render-manifest.csv`: 0 filas de datos.
- `paper/appendices/data/paper-tables-spec.csv`: 4 filas de datos.
- `paper/appendices/data/tables-evidence-manifest.csv`: 50 filas de datos.
- `paper/appendices/data/architecture-component-matrix.csv`: 50 filas de datos.
- `paper/appendices/data/selection-score-matrix.csv`: 116 filas de datos.
- `paper/appendices/data/critical-appraisal-matrix.csv`: 50 filas de datos.
- `paper/appendices/data/intake.md`: documento metodológico (20 líneas).
- `paper/appendices/data/research-question.md`: documento metodológico (15 líneas).
- `paper/appendices/data/eligibility-criteria.md`: documento metodológico (17 líneas).
- `paper/appendices/data/review-mode.md`: documento metodológico (124 líneas).
- `paper/appendices/data/review-mode.json`: documento metodológico (131 líneas).
- `paper/appendices/data/search-decomposition.md`: documento metodológico (221 líneas).
- `paper/appendices/data/search-decomposition.json`: documento metodológico (350 líneas).
- `paper/appendices/data/search-strategy.md`: documento metodológico (251 líneas).
- `paper/appendices/data/network-methodology.md`: documento metodológico (45 líneas).
- `paper/appendices/data/network-summary.md`: documento metodológico (20 líneas).
- `paper/appendices/data/network-nodes.csv`: 1266 filas de datos.
- `paper/appendices/data/network-edges.csv`: 3541 filas de datos.
- `paper/appendices/data/network-centrality.csv`: 1699 filas de datos.
- `paper/appendices/data/network-communities.csv`: 238 filas de datos.
- `paper/appendices/data/network-author-production.csv`: 283 filas de datos.
- `paper/appendices/data/network-selection-drift.csv`: 6033 filas de datos.
- `paper/appendices/data/network-coverage.json`: documento metodológico (65 líneas).
- `paper/appendices/data/network-parameters.json`: documento metodológico (26 líneas).
- `paper/appendices/data/network-provenance.csv`: 273 filas de datos.
- `paper/appendices/data/prisma-checklist.md`: documento metodológico (57 líneas).

En esta revisión, `figures-evidence-manifest.csv` registra candidatos visuales detectados en los PDFs; ese inventario no implica inclusión en el manuscrito. Una figura fuente solo debe entrar en el artículo si se inserta como figura concreta, con caption analítica y aporte claro al argumento.

Estos anexos CSV forman parte de la aportación documental del artículo y deben acompañar al manuscrito final siempre que la política editorial de la revista lo permita.

Declaraciones editoriales

- Perfil editorial operativo: `generic-common-core`.
- Estilo de citación y referencias: APA 7 aplicada al cuerpo del manuscrito, tablas, figuras y bibliografía final.
- Conflictos de interés: no se declaran conflictos de interés.
- Financiación: no se ha recibido financiación específica para la realización de este trabajo.
- Disponibilidad de datos y materiales: el corpus bibliográfico, los anexos CSV, las matrices de cribado/extracción y los activos visuales propios se conservan en el paquete editorial. Los PDFs locales se usan solo como evidencia privada de auditoría y no deben redistribuirse si su licencia no lo permite.

- Checklist de reporte de revisión sistemática: se adjunta como anexo metodológico dentro del paquete editorial para facilitar evaluación de revista.
- Cumplimiento metodológico: el manuscrito se compila desde un protocolo de revisión sistemática de literatura con trazabilidad DOI, recuperación de PDF local, extracción estructurada, evaluación de calidad y auditoría editorial determinista previa al criterio final de publicabilidad.

Referencias

- Addagada, T. C. (2026). *Semalith v1.4: A calibrated 184m safety classifier achieving state-of-the-art prompt-injection detection at 44x fewer parameters than llama-guard-3-8b* [Preprint]. arXiv. <https://arxiv.org/abs/2607.22545>
- Alizadeh, M., Samei, Z., Stetsenko, D., & Gilardi, F. (2025). *Simple prompt injection attacks can leak personal data observed by LLM agents during task execution* [Preprint]. arXiv. <https://arxiv.org/abs/2506.01055>
- An, H., Zhang, J., Du, T., Zhou, C., Li, Q., Lin, T., & Ji, S. (2025). Ipiguard: A novel tool dependency graph-based defense against indirect prompt injection in LLM agents. *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 1023-1039. <https://doi.org/10.18653/v1/2025.emnlp-main.53>
- Atul. (2026). *A multi-agent framework for explainable prompt injection detection in large language models* [Preprint]. Research Square. <https://doi.org/10.21203/rs.3.rs-10302085/v1>
- Ba, L., Li, Q., & Li, S. (2026). *CIBER: A comprehensive benchmark for security evaluation of code interpreter agents* [Preprint]. arXiv. <https://arxiv.org/abs/2602.19547>
- Balashov, A., Ponomarova, O., & Zhai, X. (2025). *Multi-stage prompt inference attacks on enterprise LLM systems* [Preprint]. arXiv. <https://arxiv.org/abs/2507.15613>
- Betser, R., Bose, S., Giloni, A., Picardi, C., Padakandla, S., & Vainshtein, R. (2026). *AgentTRIM: Tool risk mitigation for agentic AI* [Preprint]. arXiv. <https://arxiv.org/abs/2601.12449>
- Bhagwatkar, R., Kasa, K., Puri, A., Huang, G., Rish, I., Taylor, G. W., Dvijotham, K. D., & Lacoste, A. (2025). *Indirect prompt injections: Are firewalls all you need, or stronger benchmarks?* [Preprint]. arXiv. <https://arxiv.org/abs/2510.05244>
- Brooks, D., & Turner, S. (2025). An empirical evaluation of prompt injection detection and refusal-usefulness tradeoffs using the deepset/prompt-injections dataset. *Journal of Computing Innovations and Applications*, 3(2), 66-84. <https://doi.org/10.63575/cia.2025.30205>
- Chang, H., Bao, E., Luo, X., & Yu, T. (2026). *Overcoming the retrieval barrier: Indirect prompt injection in the wild for LLM systems* [Preprint]. arXiv. <https://arxiv.org/abs/2601.07072>
- Chauhan, K., & Revankar, P. (2026). *Caught in the act(ivation): Toward pre-output and multi-turn detection of credential exfiltration by LLM agents* [Preprint]. arXiv. <https://arxiv.org/abs/2606.04141>
- Chen, Y. S., Huang, S. Y., Yang, C. L., & Chen, Y. N. (2026). *TraceSafe: A systematic assessment of LLM guardrails on multi-step tool-calling trajectories* [Preprint]. arXiv. <https://arxiv.org/abs/2604.07223>
- Cunningham, H., Wei, J., Wang, Z., Persic, A., Peng, A., Abderrachid, J., Agarwal, R., Chen, B., Cohen, A., Dau, A., Dimitriev, A., Gilson, R., Howard, L., Hua, Y., Kaplan, J., Leike, J., Lin, M., Liu, C., Mikulik, V., ... Sharma, M. (2026). *Constitutional classifiers++: Efficient production-grade defenses against universal jailbreaks* [Preprint]. arXiv. <https://arxiv.org/abs/2601.04603>
- Das, D., Beurer-Kellner, L., Fischer, M., & Baader, M. (2025). *CommandSans: Securing AI agents with surgical precision prompt sanitization* [Preprint]. arXiv. <https://arxiv.org/abs/2510.08829>

- Das, D., Piet, J., Kaviani, D., Beurer-Kellner, L., Tramèr, F., & Wagner, D. (2026). *Trojan hippo: Weaponizing agent memory for data exfiltration* [Preprint]. arXiv. <https://arxiv.org/abs/2605.01970>
- Deep, P., Emmons, S., Fox, A., Bacon, K., McAllister, K., Ortiz, P., & Flautner, K. (2026). *Evaluation of prompt injection defenses in large language models* [Preprint]. arXiv. <https://arxiv.org/abs/2604.23887>
- Del Rosario, R. F., Krawiecka, K., & de Witt, C. S. (2025). *Architecting resilient LLM agents: A guide to secure plan-then-execute implementations* [Preprint]. arXiv. <https://arxiv.org/abs/2509.08646>
- Derya, K., & Sunar, B. (2026). *Revisiting jbsshield: Breaking and rebuilding representation-level jailbreak defenses* [Preprint]. arXiv. <https://arxiv.org/abs/2605.03095>
- Dingeto, H., & Leeney, W. (2026). *AgentRedBench: Dynamic redteaming and integration-aware defense for LLM agents over SaaS integrations* [Preprint]. arXiv. <https://arxiv.org/abs/2606.02240>
- Fan, L., Li, Z., Tian, Y., Wang, Y., Li, R., & Wang, X. (2026). *The granularity mismatch in agent security: Argument-level provenance solves enforcement and isolates the LLM reasoning bottleneck* [Preprint]. arXiv. <https://arxiv.org/abs/2605.11039>
- Fogel, A., Hofman, O., Cohen, E., & Vainshtein, R. (2026). *Inference-time backdoors via chat templates: From LLM supply chains to agentic system compromise* [Preprint]. arXiv. <https://arxiv.org/abs/2602.04653>
- Ge, Y. (2026). *Governance architecture for autonomous agent systems: Threats, framework, and engineering practice* [Preprint]. arXiv. <https://arxiv.org/abs/2603.07191>
- Geng, R., Wang, Y., Yin, C., Cheng, M., Chen, Y., & Jia, J. (2025). *Pisanitizer: Preventing prompt injection to long-context LLMs via prompt sanitization* [Preprint]. arXiv. <https://arxiv.org/abs/2511.10720>
- Gong, G., & Deng, Z. (2026). *PlanGuard: Defending agents against indirect prompt injection via planning-based consistency verification* [Preprint]. arXiv. <https://arxiv.org/abs/2604.10134>
- Goren, G., Katz, S., & Wolf, L. (2026). AlignTree: Efficient defense against LLM jailbreak attacks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(44), 37417-37425. <https://doi.org/10.1609/aaai.v40i44.41074>
- Gowda, I. (2026). *MEMSAD: Gradient-coupled anomaly detection for memory poisoning in retrieval-augmented agents* [Preprint]. arXiv. <https://arxiv.org/abs/2605.03482>
- Guo, W., Zeng, W., Liu, C., Jia, X., Xu, Y., Tang, L., Fang, Y., & Liu, Y. (2026). *MalSkillBench: A runtime-verified benchmark of malicious agent skills* [Preprint]. arXiv. <https://arxiv.org/abs/2606.07131>
- Gupta, A. (2025). *Can AI keep a secret? Contextual integrity verification: A provable security architecture for LLMs* [Preprint]. arXiv. <https://arxiv.org/abs/2508.09288>
- He, L., Wang, Y., Zhang, J., & Asokan, N. (2026). *Defending against adaptive prompt injection attacks via reasoning-enabled task alignment* [Preprint]. arXiv. <https://arxiv.org/abs/2606.15441>
- He, Y., Zhu, H., Li, Y., Shao, S., Yao, H., Liu, Z., & Qin, Z. (2026). *AttriGuard: Defeating indirect prompt injection in LLM agents via causal attribution of tool invocations* [Preprint]. arXiv. <https://arxiv.org/abs/2603.10749>
- Hu, Z., Wu, G., Mitra, S., Zhang, R., Sun, T., Huang, H., & Swaminathan, V. (2023). *Token-level adversarial prompt detection based on perplexity measures and contextual information* [Preprint]. arXiv. <https://arxiv.org/abs/2311.11509>

- Huang, C., Huang, X., & Fard, A. M. (2026). *Are AI-assisted development tools immune to prompt injection?* [Preprint]. arXiv. <https://arxiv.org/abs/2603.21642>
- Isbarov, J., & Kantarcioglu, M. (2026). *Bypassing AI control protocols via agent-as-a-proxy attacks* [Preprint]. arXiv. <https://arxiv.org/abs/2602.05066>
- Jamshidi, S., Dakhel, A. M., Nafi, K. W., & Khomh, F. (2025). *Semantic attacks on tool-augmented LLMs: Securing the model context protocol against descriptor-level manipulation* [Preprint]. arXiv. <https://arxiv.org/abs/2512.06556>
- Ji, Z., Wang, X., Li, Z., Ma, P., Gao, Y., Wu, D., Yan, X., Tian, T., & Wang, S. (2025). *Taxonomy, evaluation and exploitation of IPI-centric LLM agent defense frameworks* [Preprint]. arXiv. <https://arxiv.org/abs/2511.15203>
- Jia, F., Wu, T., Qin, X., & Squicciarini, A. (2025). The task shield: Enforcing task alignment to defend against indirect prompt injection in LLM agents. *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 29680-29697. <https://doi.org/10.18653/v1/2025.acl-long.1435>
- Jiang, T., Wang, Y., Liang, J., & Wang, T. (2026). *AgentLAB: Benchmarking LLM agents against long-horizon attacks* [Preprint]. arXiv. <https://arxiv.org/abs/2602.16901>
- Kang, M., Xiang, C., Kariyappa, S., Xiao, C., Li, B., & Suh, E. (2025). *Mitigating indirect prompt injection via instruction-following intent analysis* [Preprint]. arXiv. <https://arxiv.org/abs/2512.00966>
- Karanjai, R., Lu, Y., Madhavarao, H. H., Xu, L., & Shi, W. (2026). *Context contamination in LLM analysis of network security logs: Poison with passive prompt injection and mitigation evaluation* [Preprint]. arXiv. <https://arxiv.org/abs/2607.14493>
- Kasundra, J., Praharaj, A., Surana, S., Chodisetty, L. S., Sharma, S., Verma, A., Bhardwaj, A., Kanhar, D., Bhagat, A., Slimi, K., Subramanian, S., Madhusudhan, S. T., Chenna, R. P., & Sunkara, S. (2025). *AprielGuard* [Preprint]. arXiv. <https://arxiv.org/abs/2512.20293>
- Kim, J., Choi, W., Kang, T., Kim, Y., & Lee, B. (2026). *DualView: Preventing indirect prompt injection in personal AI agents* [Preprint]. arXiv. <https://arxiv.org/abs/2607.03821>
- Kim, M., Parmar, M., Wallis, P., Miculicich, L., Jung, K., Dvijotham, K. D., Le, L. T., & Pfister, T. (2026). *CausalArmor: Efficient indirect prompt injection guardrails via causal attribution* [Preprint]. arXiv. <https://arxiv.org/abs/2602.07918>
- Kim, W., & Yoo, H. K. (2026). *SecureMCP: A policy-enforced LLM data access framework for AIoT systems via model context protocol* [Preprint]. arXiv. <https://arxiv.org/abs/2605.05260>
- Kitchenham, B., & Charters, S. (2007). *Guidelines for performing systematic literature reviews in software engineering* (EBSE Technical Report EBSE-2007-01). Keele University and Durham University. <https://madeyski.e-informatyka.pl/download/Kitchenham07.pdf>
- Kolluri, A., Sharma, R., Costa, M., Köpf, B., Nießen, T., Russinovich, M., Tople, S., & Zanella-Béguelin, S. (2026). *Optimizing agent planning for security and autonomy* [Preprint]. arXiv. <https://arxiv.org/abs/2602.11416>
- Kumar, V. (2026). Shieldllm: A hybrid adversarial prompt injection detection framework for securing large language models. *International Journal of Creative and Open Research in Engineering and Management*, 02(04), 1-9. <https://doi.org/10.55041/ijcope.v2i4.463>
- Lan, Q., & Kaul, A. (2026). *The cognitive firewall: Securing browser based AI agents against indirect prompt injection via hybrid edge cloud defense* [Preprint]. arXiv. <https://arxiv.org/abs/2603.23791>

- Lavanya, Y., Reshma, P., Aditya, G. J., Raja, M. A. D., & Lakshmi, B. M. (2026). LLM prompt injection detection firewall. *International Journal of Engineering Research and Science & Technology*, 22(2(1)), 66-72. [https://doi.org/10.62643/ijerst.2026.v22.n2\(1\).pp66-72](https://doi.org/10.62643/ijerst.2026.v22.n2(1).pp66-72)
- Li, F. (2026). *OpenClaw PRISM: A zero-fork, defense-in-depth runtime security layer for tool-augmented LLM agents* [Preprint]. arXiv. <https://arxiv.org/abs/2603.11853>
- Li, H., & Liu, X. (2024). *InjecGuard: Benchmarking and mitigating over-defense in prompt injection guardrail models* [Preprint]. arXiv. <https://arxiv.org/abs/2410.22770>
- Li, H., Yang, Y., Suh, G. E., Zhang, N., & Xiao, C. (2026). *ReasAlign: Reasoning enhanced safety alignment against prompt injection attack* [Preprint]. arXiv. <https://arxiv.org/abs/2601.10173>
- Li, R., Chen, M., Hu, C., Chen, H., Xing, W., & Han, M. (2024). *GenTel-safe: A unified benchmark and shielding framework for defending against prompt injection attacks* [Preprint]. arXiv. <https://arxiv.org/abs/2409.19521>
- Liang, Z., Hu, T., Chen, Z., & Tang, M. (2025). *Cognitive control architecture (CCA): A lifecycle supervision framework for robustly aligned AI agents* [Preprint]. arXiv. <https://arxiv.org/abs/2512.06716>
- Lin, J., Zhou, Z., Zheng, Z., Liu, S., Xu, T., Chen, Y., & Chen, E. (2026). *VIGIL: Defending LLM agents against tool stream injection via verify-before-commit* [Preprint]. arXiv. <https://arxiv.org/abs/2601.05755>
- Lin, Z., Niu, Z., Ji, H., & Gao, H. (2026). *Guaranteed jailbreaking defense via disrupt-and-rectify smoothing* [Preprint]. arXiv. <https://arxiv.org/abs/2605.10582>
- Liu, H., Xu, D., Ma, Q., Xu, S., & Qiu, D. (2026). *Memory poisoning propagation and repair mechanism in multi-agent collaborative environments* [Preprint]. Preprints.org. <https://doi.org/10.20944/preprints202602.1188.v1>
- Liu, M., Zha, Y., & Chen, M. (2026). *Piiguard: Mitigating PII harvesting under adversarial sanitization* [Preprint]. arXiv. <https://arxiv.org/abs/2605.03129>
- Louck, Y. (2026). *Securing LLM-agent long-term memory against poisoning: Non-malleable, origin-bound authority with machine-checked guarantees* [Preprint]. arXiv. <https://arxiv.org/abs/2606.24322>
- Luo, J., & Zhang, C. (2026). *Real user instruction: Black-box instruction authentication middleware against indirect prompt injection* [Preprint]. Preprints.org. <https://doi.org/10.20944/preprints202603.1023.v1>
- Ma, X., Li, T., Xiao, C., Yu, Z., Zhang, N., & Vorobeychik, Y. (2026). *AutoDojo: Adaptive black-box attacks reveal the limits of IPI defenses and task-specification effects in LLM agents* [Preprint]. arXiv. <https://arxiv.org/abs/2606.15057>
- Maiorano, A. C. (2026). *Evaluating prompt injection defenses for educational LLM tutors: Security-usability-latency trade-offs* [Preprint]. arXiv. <https://arxiv.org/abs/2605.06669>
- Marinelli, R., Pichlmeier, J., & Bisztray, T. (2025). *Harnessing chain-of-thought metadata for task routing and adversarial prompt detection* [Preprint]. arXiv. <https://arxiv.org/abs/2503.21464>
- Meng, M., & Zhang, Z. (2025). *A biosecurity agent for lifecycle LLM biosecurity alignment* [Preprint]. bioRxiv. <https://doi.org/10.1101/2025.09.17.676717>
- Mohanty, S. K., Patel, R., Yuvaraj, K., Chaudhary, J., & Singhania, D. (2026). *TriShieldRAG: A three-ring defense-in-depth framework against knowledge corruption in retrieval-augmented generation* [Preprint]. arXiv. <https://arxiv.org/abs/2607.23838>
- Mudryi, M., Chaklosh, M., & Wójcik, G. (2025). *The hidden dangers of browsing AI agents* [Preprint]. arXiv. <https://arxiv.org/abs/2505.13076>

- Narisetty, P., Kore, S. N. B., Kattamanchi, U. K. R., & Kumarapu, J. (2026). *Adaptive evaluation of out-of-band defenses against prompt injection in LLM agents* [Preprint]. arXiv. <https://arxiv.org/abs/2606.26479>
- Neves, P. R. F., Filho, E. R. D. C., Falsetti, P. H. E., Pavan, J. V., Degaspari, I., Laturrague, H. V., Laturrague, P. V., Dias, G. N., Berto, M. W. P., & Von Atzingen, G. V. (2026). *GuardNet: Ensemble strategies of shallow neural networks for robust prompt injection and jailbreak detection* [Preprint]. arXiv. <https://arxiv.org/abs/2606.05566>
- Niu, P., Qu, W., Gu, S., Shi, T., Li, Y., Tawaha, A., Alzahrani, H., Siu, V., Li, B., Wang, C., Zhang, J., Alomair, B., Jin, M., Chen, M., Wang, C., Spanos, C., & Song, D. (2026). *Understanding and evaluating claw-like agent security through a computer-systems lens* [Preprint]. arXiv. <https://arxiv.org/abs/2606.30755>
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., ... Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*, 372, n71. <https://doi.org/10.1136/bmj.n71>
- Pai, A. (2026). *PARSE: Provenance-aware retrieval sanitization for professional domain LLM agents* [Preprint]. arXiv. <https://arxiv.org/abs/2606.17467>
- Pan, S., Sun, X., Zhang, T., Liao, D., Yang, K., & Xing, Z. (2026). *SkillGuard: A permission-centric framework for agent skill security* [Preprint]. arXiv. <https://arxiv.org/abs/2606.03024>
- Potluri, M. (2026a). *A layered guardrail architecture for agentic AI in cloud-native DevOps pipelines: Design, implementation, and empirical evaluation on live AWS infrastructure* [Preprint]. Research Square. <https://doi.org/10.21203/rs.3.rs-9890341/v1>
- Potluri, M. (2026b). *Why single-layer LLM guardrails fail: A dual-detection pattern on AWS bedrock* [Preprint]. Research Square. <https://doi.org/10.21203/rs.3.rs-9707431/v1>
- Ramakrishnan, B., & Balaji, A. (2025). *Securing AI agents against prompt injection attacks* [Preprint]. arXiv. <https://arxiv.org/abs/2511.15759>
- Rassul, Y. H., & Rashid, T. A. (2026). *AgentShield: Deception-based compromise detection for tool-using LLM agents* [Preprint]. arXiv. <https://arxiv.org/abs/2605.11026>
- Rethlefsen, M. L., Kirtley, S., Waffenschmidt, S., Ayala, A. P., Moher, D., Page, M. J., & Koffel, J. B. (2021). PRISMA-S: An extension to the PRISMA statement for reporting literature searches in systematic reviews. *Systematic Reviews*, 10, Article 39. <https://doi.org/10.1186/s13643-020-01542-z>
- Saravi, B., Singh, D. D., Schorn, L., Vollmer, A., Sproll, C., Kübler, N., & Schrader, F. (2026). *Image-embedded prompt injection vulnerability of vision-language models in dental radiology: a cross-vendor attack-defense evaluation* [Preprint]. Research Square. <https://doi.org/10.21203/rs.3.rs-9932271/v1>
- Schlichtkrull, M. (2025). *Attacks by content: Automated fact-checking is an AI security issue* [Preprint]. arXiv. <https://arxiv.org/abs/2510.11238>
- Shah, H. (2026). *LogJack: Indirect prompt injection through cloud logs against LLM debugging agents* [Preprint]. arXiv. <https://arxiv.org/abs/2604.15368>
- Shao, Z., Fleming, C., & Baluta, T. (2026). *Cordyceps: Covert control attacks on LLMs via data poisoning* [Preprint]. arXiv. <https://arxiv.org/abs/2605.26595>
- Sharma, M., Tong, M., Mu, J., Wei, J., Kruthoff, J., Goodfriend, S., Ong, E., Peng, A., Agarwal, R., Anil, C., Askell, A., Bailey, N., Benton, J., Blumke, E., Bowman, S. R., Christiansen, E., Cunningham, H., Dau, A., Gopal, A., ... Perez, E. (2025). *Constitutional classifiers: Defending against universal jailbreaks across thousands of hours of red teaming* [Preprint]. arXiv. <https://arxiv.org/abs/2501.18837>

- Sharma, S. (2026). *Containment over detection: Cryptographic boundary enforcement for prompt injection defense in agentic LLM systems* [Preprint]. Research Square. <https://doi.org/10.21203/rs.3.rs-10196595/v1>
- Shehmir, S., & Kashef, R. (2026). RoLLMRec: a robust LLM-based recommender system for defending against shilling and prompt injection attacks. *Frontiers in Computer Science*, 8, Article 1735253. <https://doi.org/10.3389/fcomp.2026.1735253>
- Shi, S., Wang, X., Zhang, C., Li, H., Lou, W., Hou, T., Vorobeychik, Y., Zhang, C., & Zhang, N. (2026). *Think twice before you act: Protecting LLM agents against tool description poisoning via isolated planning* [Preprint]. arXiv. <https://arxiv.org/abs/2606.20922>
- Shi, T., Zhu, K., Wang, Z., Jia, Y., Cai, W., Liang, W., Wang, H., Alzahrani, H., Lu, J., Kawaguchi, K., Alomair, B., Zhao, X., Wang, W. Y., Gong, N., Guo, W., & Song, D. (2025). *PromptArmor: Simple yet effective prompt injection defenses* [Preprint]. arXiv. <https://arxiv.org/abs/2507.15219>
- Shih, Y. K., & Kang, Y. K. (2025). *Design and implementation of a secure RAG-enhanced AI chatbot for smart tourism customer service: Defending against prompt injection attacks – a case study of hsinchu, taiwan* [Preprint]. arXiv. <https://arxiv.org/abs/2509.21367>
- SingGuard Team. (2026). *SingGuard-NSFA: Extensible guardrails for agentic AI via generative reasoning and real-time classification* [Preprint]. arXiv. <https://arxiv.org/abs/2607.13081>
- Singh, B. P. (2026). Securing the boundary: Trust context separation in privileged AI agent systems. *Computer Fraud and Security*, 998-1009. <https://doi.org/10.52710/cfs.1012>
- Singh, P., Gupta, K., & Singh, P. (2026). TRACER-AI: A multi-layer explainable framework for prompt injection, agent goal hijacking, and tool misuse detection in agentic AI systems. *International Journal for Research in Applied Science and Engineering Technology*, 14(7), 1793-1804. <https://doi.org/10.22214/ijraset.2026.84360>
- Singh, V. (2026). Zero-trust architecture patterns for securing AI-driven IP litigation support systems. *American Journal of Technology*, 5(5), 42-57. <https://doi.org/10.58425/ajt.v5i5.539>
- Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, 104, 333-339. <https://doi.org/10.1016/j.jbusres.2019.07.039>
- Sunil, B. D., Sinha, I., Maheshwari, P., Todmal, S., Mallik, S., & Mishra, S. (2026). *Memory poisoning attack and defense on memory based LLM-agents* [Preprint]. arXiv. <https://arxiv.org/abs/2601.05504>
- Tan, W. M., Lim, C., & Lee, J. (2026). Frequency-domain characterization of memory poisoning propagation in multi-agent collaborative systems. *Frontiers in Applied Physics and Mathematics*, 3(2), 44-52. <https://doi.org/10.71465/fapm760>
- Thienpreecha, P. (2026). *CrackedPDFs: A controlled benchmark for hidden prompt injection in PDFs* [Preprint]. arXiv. <https://arxiv.org/abs/2607.19396>
- Thornton, S. (2026). *TRYLOCK: Defense-in-depth against LLM jailbreaks via layered preference and representation engineering* [Preprint]. arXiv. <https://arxiv.org/abs/2601.03300>
- Torrielli, F., Locci, S., Rapp, A., & Di Caro, L. (2026). Exploiting large language models in peer review: indirect prompt injection attacks and integrity probes. *Scientometrics*, 131(7), 4889-4951. <https://doi.org/10.1007/s11192-026-05695-x>
- Wang, C., Zhang, F., Zhang, J., Zhang, Z., Wang, Y., Huang, L., Gao, J., Chen, Z., & Lim, W. Y. B. (2026). *ICON: Indirect prompt injection defense for agents based on inference-time correction* [Preprint]. arXiv. <https://arxiv.org/abs/2602.20708>
- Wang, H., Xiao, Z., Zhang, Y., Poskitt, C. M., & Sun, J. (2026). *SafeClaw-r: Towards safe and secure multi-agent personal assistants* [Preprint]. arXiv. <https://arxiv.org/abs/2603.28807>

- Wang, P., Li, Y., & Tian, Y. (2026). *Aligning provenance with authorization: A dual-graph defense for LLM agents* [Preprint]. arXiv. <https://arxiv.org/abs/2605.26497>
- Wang, T., & Li, H. (2025). *OpenGuardrails: A configurable, unified, and scalable guardrails platform for large language models* [Preprint]. arXiv. <https://arxiv.org/abs/2510.19169>
- Wei, Y., Xu, Y., Li, Z., Shen, X., & Ji, S. (2026). *Beyond input guardrails: Reconstructing cross-agent semantic flows for execution-aware attack detection* [Preprint]. arXiv. <https://arxiv.org/abs/2603.04469>
- Xie, Y., Luo, M., Liu, Z., Zhang, Z., Zhang, K., Liu, Y., Li, Z., Chen, P., Wang, S., & She, D. (2025). *Red-teaming coding agents from a tool-invocation perspective: An empirical security assessment* [Preprint]. arXiv. <https://arxiv.org/abs/2509.05755>
- Xie, Y., Yi, J., Shao, J., Curl, J., Lyu, L., Chen, Q., Xie, X., & Wu, F. (2023). Defending ChatGPT against jailbreak attack via self-reminders. *Nature Machine Intelligence*, 5(12), 1486-1496. <https://doi.org/10.1038/s42256-023-00765-8>
- Xing, W., Qi, Z., Qin, Y., Li, Y., Chang, C., Yu, J., Lin, C., Xie, Z., & Han, M. (2025). *MCP-guard: A multi-stage defense-in-depth framework for securing model context protocol in agentic AI* [Preprint]. arXiv. <https://arxiv.org/abs/2508.10991>
- Yang, T., Li, J., Nian, Y., Dong, S., Xu, R., Rossi, R., Ding, K., & Zhao, Y. (2026). *No attacker needed: Unintentional cross-user contamination in shared-state LLM agents* [Preprint]. arXiv. <https://arxiv.org/abs/2604.01350>
- Yeduguru, K. R. (2026). *Adversarial threats to AI-augmented security operations: A systematic survey* [Preprint]. Research Square. <https://doi.org/10.21203/rs.3.rs-9602180/v1>
- Ying, Z., Wang, H., Liu, J., Zou, Q., Liu, A., Yang, J., Yang, Y., & Liu, X. (2026). *AgentVisor: Defending LLM agents against prompt injection via semantic virtualization* [Preprint]. arXiv. <https://arxiv.org/abs/2604.24118>
- Yu, Q., Cheng, X., & Liu, C. (2026). *Defense against indirect prompt injection via tool result parsing* [Preprint]. arXiv. <https://arxiv.org/abs/2601.04795>
- Yuan, Z., Chen, Z., Xiang, Z., Bastian, N. D., Hashemi, S. H., Xiao, C., Guo, W., & Li, B. (2026). *ShieldNet: Network-level guardrails against emerging supply-chain injections in agentic systems* [Preprint]. arXiv. <https://arxiv.org/abs/2604.04426>
- Yung, C., Huang, H., Leckie, C., & Erfani, S. (2025). *Geometry-guided adversarial prompt detection via curvature and local intrinsic dimension* [Preprint]. arXiv. <https://arxiv.org/abs/2503.03502>
- Zha, M., & Wang, X. (2026). *Autonomous LLM agent worms: Cross-platform propagation, automated discovery and temporal re-entry defense* [Preprint]. arXiv. <https://arxiv.org/abs/2605.02812>
- Zhang, K., Su, Z., Chen, P. Y., Bertino, E., Zhang, X., & Li, N. (2025). *LLM agents should employ security principles* [Preprint]. arXiv. <https://arxiv.org/abs/2505.24019>
- Zhang, T., Xu, Y., Wang, J., Guo, K., Xu, X., Xiao, B., Guan, Q., Fan, J., Liu, J., Liu, Z., & Hu, H. (2026). *AgentSentry: Mitigating indirect prompt injection in LLM agents via temporal causal diagnostics and context purification* [Preprint]. arXiv. <https://arxiv.org/abs/2602.22724>
- Zhao, L., Bhaskar, A., & Dobriban, E. (2026). *LivePI: More realistic benchmarking of agents against indirect prompt injection* [Preprint]. arXiv. <https://arxiv.org/abs/2605.17986>
- Zhao, W., Li, Z., Zhang, P., & Sun, J. (2026). *ClawGuard: A runtime security framework for tool-augmented LLM agents against indirect prompt injection* [Preprint]. arXiv. <https://arxiv.org/abs/2604.11790>

- Zheng, D., & Li, C. (2024). Behavior-level jailbreak resistance via multi-stage refusal + utility preservation. *Journal of Advanced Computing Systems*, 4(1), 83-99. <https://doi.org/10.69987/jacs.2024.40107>
- Zhou, Y., Wang, X., Ma, P., Xue, Z., Wang, Z., & Wang, S. (2026). *From shield to target: Denial-of-service attacks on LLM-based agent guardrails* [Preprint]. arXiv. <https://arxiv.org/abs/2606.14517>
- Zhu, K., Yang, X., Wang, J., Guo, W., & Wang, W. Y. (2025). *MELON: Provable defense against indirect prompt injection attacks in AI agents* [Preprint]. arXiv. <https://arxiv.org/abs/2502.05174>
- Zhu, W., Dong, X., Chen, X., Cai, R., Qiu, P., Wang, Z., Frunza, O., Tang, S., Gu, J., & Wang, Y. (2026). *Your agent is more brittle than you think: Uncovering indirect injection vulnerabilities in agentic LLMs* [Preprint]. arXiv. <https://arxiv.org/abs/2604.03870>
- Zou, Z., Guo, S., Zhan, Q., Zhao, L., Li, S., Li, Q., Xu, K., Xu, M., & Liu, Z. (2026). *Blind gods and broken screens: Architecting a secure, intent-centric mobile agent operating system* [Preprint]. arXiv. <https://arxiv.org/abs/2602.10915>