

De la gobernanza a la evidencia: agentes que ejecutan herramientas por MCP en un laboratorio virtualizado, leídos con ISO/IEC 42001 y 27090

Ricardo Martínez

Cloud Security & IA Aplicada — NullGhost Research

Octubre 2026

Resumen

Un equipo pequeño que usa un agente de IA para conducir herramientas de seguridad dentro de una máquina virtual tiene dos preguntas encima: cómo gobernar ese uso de IA y cómo sustentar lo que el agente afirma. Este texto no mide la eficacia del agente ni certifica nada. Describe la arquitectura observada de un laboratorio propio —un operador, un modelo de lenguaje, un conector MCP, una VM Kali en un host Mac y herramientas de reconocimiento sobre objetivos propios y autorizados— y la lee contra el alcance público de ISO/IEC 42001:2023 (gestión de IA) e ISO/IEC 27090:2026 (amenazas a sistemas de IA). Sostiene tres distinciones que el laboratorio hizo evidentes: ejecutar un pentest *con* IA no es evaluar la seguridad *de* la IA; una máquina virtual no es aislamiento; y la trazabilidad de la evidencia no es conformidad con una norma. Propone una tabla de gobernanza a evidencia, con el estado real de cada actividad (casi todo pendiente), y registra las limitaciones medidas: diferencias de navegador, errores de automatización y fallo silencioso de herramientas. No se leyó el texto íntegro de ninguna de las dos normas.

Nota de proceso. La operación del laboratorio, la búsqueda y la redacción tuvieron asistencia de IA. Las afirmaciones sobre las normas vienen de las fuentes citadas, no del modelo. Cada fuente lleva una etiqueta: [P] primaria, leída directamente; [S] secundaria, resumen o cobertura de un tercero que no se pudo contrastar con el original; [A] afirmación de una parte interesada; [O] observación propia del laboratorio, que es dato de una sola operación, no evidencia generalizable. El texto marca en prosa si cada afirmación es observación, inferencia, propuesta o referencia normativa. Las pruebas se corrieron solo contra objetivos propios de NullGhost y con autorización; la seguridad de terceros y cualquier divulgación se tratan por separado y no forman parte de este documento. Corte de la investigación: 19 de octubre de 2026.

1. Problema y alcance

NullGhost es un equipo pequeño que usa IA de dos maneras que conviene no confundir. Una es un asistente acotado dentro de su plataforma (modelo local, sin herramientas, sin datos de clientes). La otra, la de este texto, es un agente que conduce herramientas de

seguridad dentro de una máquina virtual para revisar objetivos propios. Las dos caen bajo el gobierno de IA de la organización, pero plantean riesgos distintos.

La pregunta de investigación es cómo vincular el gobierno de esos usos de IA con la evaluación técnica y con la trazabilidad de lo que el agente hace y afirma. No es una pregunta de eficacia (“¿encuentra más el agente?”), sino de gobernanza y evidencia (“¿se puede reconstruir qué se autorizó, qué se ejecutó y qué cambió?”). Este documento responde a la segunda y deja la primera fuera.

2. Qué se sabe de las normas

La ficha pública de ISO/IEC 27090 la describe como una guía para atender amenazas y compromisos de seguridad contra sistemas de IA a lo largo de su ciclo de vida, y cómo detectarlos y mitigarlos. Según el sitio ISO27k cubre al menos una docena de escenarios de ataque deliberado —envenenamiento de datos y de modelos, inversión y robo de modelos, entre otros— y complementa a ISO/IEC 27001 y 27002 [1, 2]. La página oficial de ISO no se pudo abrir desde nuestro entorno (respondió con acceso bloqueado), así que la

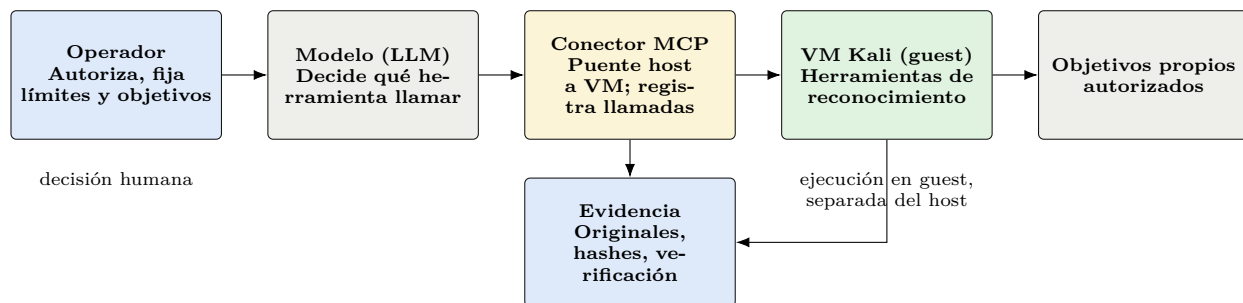


Figura 1: Arquitectura observada del laboratorio (observación propia). El operador y el modelo deciden; el conector MCP media y registra; las herramientas corren en la VM; la evidencia se guarda con sus huellas y se verifica aparte. Separar host y guest importa: no toda ejecución ocurre donde parece.

fecha de publicación de 2026 no está reverificada en la fuente oficial.

ISO/IEC 42001:2023 define cómo establecer, mantener y mejorar un sistema de gestión de IA en una organización, con cláusulas auditables (4 a 10), una evaluación de impacto del sistema de IA (6.1.4) y un Anexo A de controles de referencia; es certificable de forma voluntaria [3]. Que 27090, por ser guía, no defina requisitos certificables es una inferencia nuestra, a confirmar con el texto. La estructura de cláusulas y del Anexo A proviene de catálogos públicos de terceros, no de la norma leída [4].

3. Arquitectura observada

El laboratorio (Figura 1) tiene cinco piezas: un **operador** humano que autoriza y fija alcance; un **modelo de lenguaje** que, turno a turno, decide qué herramienta invocar; un **conector MCP** que traduce esas decisiones en llamadas y las registra [5]; una **VM Kali** sobre un host Mac donde corren las herramientas; y un **almacén de evidencia** con los originales y sus huellas. Las herramientas son de reconocimiento y revisión pasiva o de bajo impacto (consultas HTTP, identificación de tecnologías, plantillas de hallazgos); los objetivos fueron activos propios de NullGhost, con autorización.

Dos hechos de arquitectura resultaron relevantes para la gobernanza (observación). Primero, el conector separa el host del guest: una orden puede resolverse en el host o dentro de la VM, y confundir dónde se ejecutó contamina la evidencia. Segundo, el MCP deja un registro de llamadas que es, en sí mismo, parte de la trazabilidad: qué se pidió, cuándo y con qué argumentos.

4. Método

El método reutiliza el protocolo de evidencia que NullGhost publicó como manual [6]: autorización y límites por escrito antes de tocar nada; un plan de prueba con el efecto esperado declarado de antemano; controles que distinguen un resultado *observado* de uno *reproducido* y de uno *verificado*; resultados negativos escritos como “no observado bajo estas condiciones”, nunca como “no es vulnerable”; verificación independiente por un segundo paso sin el razonamiento original; e integridad por huellas (hash) de cada archivo, más un acta de cierre con los pendientes.

Una regla gobernó todo: ninguna afirmación se cree en la misma pasada que la produjo. El contexto histórico —lo que el agente “ya sabía” de corridas anteriores— se trata como pista, no como hecho, porque un modelo de lenguaje no distingue entre “esto ya se confirmó” y “puedo reclamarlo como propio” [7].

5. Resultados y limitaciones medidas

Los resultados concretos sobre objetivos propios no se detallan aquí porque este texto es sobre el método, no un reporte de hallazgos. Lo que sí es resultado publicable son las limitaciones que el laboratorio hizo medibles (observación):

- **Fallo silencioso.** Una herramienta puede terminar sin error aparente y sin haber hecho nada útil. El riesgo mayor no es el error visible sino el que no se nota; por eso el arnés debe medir *liveness*, no solo ausencia de excepción.
- **Disponible no es ejecutado.** Que el agente pueda llamar una herramienta no significa que la haya llamado en esa corrida. Una técnica apareció en una pasada y desapareció en la siguiente

contra el mismo objetivo, sin cambios en el entorno [7]. Las capacidades de alto valor y bajo riesgo de falso positivo conviene sacarlas de la decisión del modelo y volverlas un barrido determinista.

- **Diferencias de navegador y errores de automatización.** El mismo objetivo se vio distinto según el cliente HTTP o el navegador usado; parte de lo que parecía hallazgo era artefacto de la herramienta. Se registran como limitación, no como resultado.
- **Verificar al verificador.** Un verificador que compara contra su propia lista de verdad escrita a mano queda en verde para siempre sin medir nada. El segundo paso también se audita.

6. De la gobernanza a la evidencia

La Tabla 1 es una propuesta de NullGhost para enlazar cada objetivo propio con una actividad, su evidencia y su estado real. No es un mapa normativo ni una lista de controles atribuida a una ISO; la columna de cláusula, aplicabilidad y justificación se añadirá tras leer las normas completas. El estado honesto es que casi todo está pendiente: solo la evidencia piloto y el cierre de sesión existen hoy.

7. Discusión

Tres confusiones conviene no cometer. **Pentest con IA no es evaluar la IA.** Usar un agente para revisar un sitio produce, a lo sumo, evidencia sobre ese sitio; no dice nada sobre la seguridad del propio agente (sus datos, su modelo, sus entradas, sus permisos y sus herramientas), que exige pruebas específicas y es justamente de lo que trata 27090. **Virtualización no es aislamiento.** Una VM reduce el radio de impacto, pero compartir host, red y portapapeles deja superficie; afirmar “está aislado” sin probar la frontera es aseguramiento falso. **Trazabilidad no es conformidad.** Guardar llamadas, archivos y hashes permite reconstruir qué pasó, lo que es necesario pero no suficiente para cumplir una norma: la conformidad exige el sistema de gestión, la lectura del texto y un organismo acreditado.

8. Limitaciones y trabajo futuro

- No se leyó el texto íntegro de 42001 ni de 27090. Su alcance y estructura vienen de fuentes secun-

darias; la fecha de 27090 no está reverificada en la fuente oficial.

- Los resultados del laboratorio son de una sola operación sobre objetivos propios. No hay comparación controlada ni medición de eficacia; las cifras de corridas describen el pasado, no garantizan cobertura.
- La tabla de gobernanza es propuesta sin implementar ni evaluar. La columna normativa está pendiente.
- Trabajo futuro: leer las ediciones completas y verificar la correspondencia por cláusula; correr el método sobre un segundo laboratorio para separar lo que es del método de lo que es del operador; y probar la frontera host-guest de forma explícita.

9. Conclusión

Para un equipo pequeño, el valor de conducir herramientas por MCP desde un agente no está en que encuentre más, sino en que deja un rastro que otro puede revisar. Ese rastro habilita la gobernanza, pero no la sustituye, y no debe confundirse con evaluar la seguridad de la propia IA. La conclusión proporcional a los datos es modesta: el laboratorio es trazable y el método es reproducible; la conformidad, la eficacia y el aislamiento siguen sin demostrarse, y el texto lo dice en cada afirmación.

Referencias

- [1] ISO27k Forum. *ISO/IEC 27090: guidance for addressing security threats and compromises to artificial intelligence systems*. <https://www.iso27001security.com/html/27090> [S]
- [2] ISO27k Forum. *ISO/IEC 27090 AI security standard published*. <https://www.iso27001security.com/html/27090> [S]
- [3] ISO/IEC 42001:2023, *Information technology — Artificial intelligence — Management system*; ficha del catálogo oficial. <https://www.iso.org/standard/42001> [S]
- [4] Resúmenes públicos de la estructura de ISO/IEC 42001 (cláusulas 4–10, 6.1.4 y Anexo A: 9 objetivos, 38 controles). Verificar contra la norma. [S]
- [5] Anthropic. *Model Context Protocol: specification*. <https://modelcontextprotocol.io> [P]
- [6] NullGhost Research. *Manual de buenas prácticas de IA (guía y manual de trabajo, v0.4)*. <https://nullghost.io/ia-normas.html> [P]

Tabla 1: Propuesta de gobernanza a evidencia (propuesta propia; no es mapa normativo). Estado al 19 de octubre de 2026.

Objetivo propio	Actividad propuesta	Evidencia esperada	Verificación	Estado
Delimitar usos de IA	Inventario de modelos, conectores y datos	Alcance aprobado y responsables	Revisión con dueño del proceso	Pendiente
Controlar la delegación	Permisos mínimos y límites por herramienta	Configuración y autorizaciones fechadas	Intentos permitidos y denegados en laboratorio	Pendiente
Gestionar riesgos	Registro de amenazas y tratamientos	Registro con responsable y fecha	Revisión por riesgo concreto	Pendiente
Proteger el flujo técnico	Pruebas de entradas no confiables y de exposición	Casos sintéticos y resultados	Reproducción con controles	Pendiente
Sustentar afirmaciones	Enlazar prueba, resultado y limitación	Archivos originales y hashes	Revisión independiente	Piloto; cobertura por revisar
Cerrar y mejorar	Registrar diferencias, fallos y cambios	Acta de cierre y pendientes	Retest tras cambios	Cierre disponible

- [7] NullGhost Research. *Cuando el agente decide: diez fallos de diseño en agentes de seguridad ofensiva autónomos* (septiembre de 2026). <https://nullghost.io/research> [P]
- [8] NIST. *AI 100-2e2025: Adversarial Machine Learning*, marzo de 2025. <https://doi.org/10.6028/NIST.AI.100-2e2025> [S]
- [9] MITRE. *ATLAS, Adversarial Threat Landscape for AI Systems*. <https://atlas.mitre.org> [S]
- [10] OWASP GenAI Security Project. *OWASP Top 10 for LLM Applications 2025*. <https://genai.owasp.org> [S]