

I CONGRESO · INTELIGENCIA ARTIFICIAL · 2026

I Congreso | IA LATAM

El encuentro de la comunidad de Inteligencia Artificial de
Latinoamérica.



30 – 31 de julio de
2026



100% Online



comunidadialatam.org



IA LATAM · Comunidad

Congreso AI LATAM



CHARLA

Pentesting con IA

● Camila Casas · Pentester / Instructora · Freelance · Chile



IA LATAM · Comunidad

Congreso AI LATAM

01

SECCIÓN

Fundamentos del pentesting

Qué es un pentest tradicional, sus fases clásicas y por qué no escala.



 AGENDA

Lo que veremos hoy

1

Introducción y contexto

El pentest tradicional y sus límites

2

Por qué IA

Drivers de la automatización inteligente

3

Ciclo del pentest con IA

Reconocimiento, escaneo y explotación

4

Herramientas reales

Frameworks y agentes en producción

5

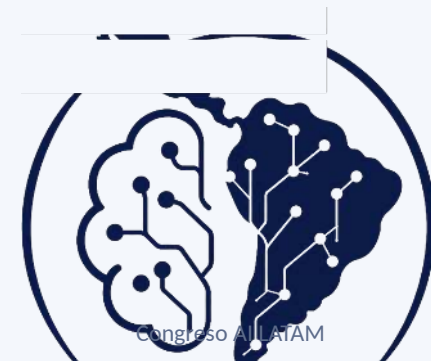
Arquitectura de pipeline

Cómo se integra todo

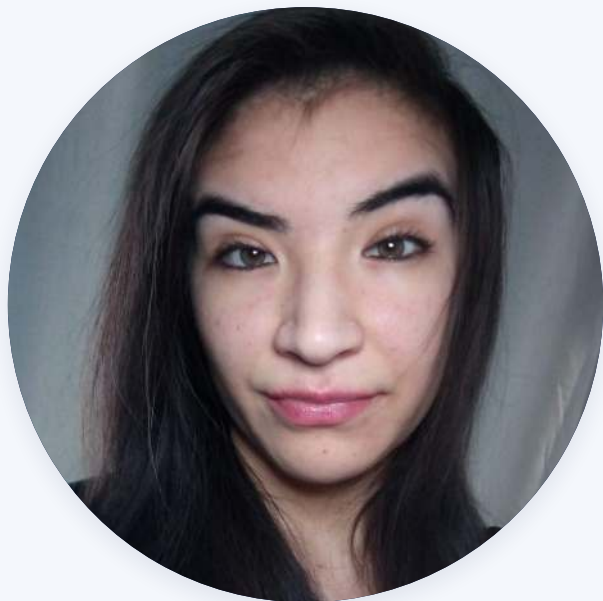
6

Laboratorio práctico con Hashlock

Uso práctico con DeFiVulnLabs



Quién te acompaña hoy



Camila Casas

Pentester & Instructora en Ciberseguridad Ofensiva • Freelance • Chile

Pentester y desarrolladora de software con más de 7 años en ciberseguridad ofensiva: ethical hacking en web, APIs, apps móviles (Android/iOS), cloud, código fuente, ERP (SAP) e inteligencia artificial.

¿Qué es un pentest?

Simula ataques reales para hallar vulnerabilidades antes que un atacante. Su metodología clasica sigue estas fases, de principio a fin:

- 1. Planificacion: define el alcance y las reglas del compromiso
- 2. Reconocimiento: mapea la superficie de ataque (OSINT, subdominios, activos)
- 3. Escaneo: identifica servicios, puertos y versiones expuestas
- 4. Enumeracion: profundiza en usuarios, permisos y configuraciones
- 5. Explotacion: valida las vulnerabilidades con exploits reales
- 6. Post-explotacion: evalua impacto y movimiento lateral
- 7. Reporte: documenta hallazgos y remediacion

Reconocimiento: manual vs. con IA



Reconocimiento manual

- OSINT manual y búsquedas dispersas
- Correlación de subdominios y leaks hecha a mano
- Horas de trabajo para mapear la superficie de ataque



Reconocimiento con IA

- OSINT impulsado por LLM
- Correlación automática de subdominios, leaks y huellas digitales
- Modelos que clasifican activos por exposición y criticidad de negocio



Fuzzing inteligente con IA

Los modelos generativos crean variantes de entrada diseñadas para maximizar la probabilidad de encontrar un comportamiento inesperado.

- Entrada base → mutación guiada por IA → ejecución contra el objetivo
- Análisis de respuesta → refinamiento continuo del payload
- Ciclos más cortos y mayor cobertura de casos límite que los diccionarios estáticos tradicionales



Entrada → Mutación IA → Ejecución → Análisis → Refinamiento



Explotación asistida por agentes de IA



Agentes autónomos (LLM-driven)

Encadenan comandos, interpretan salidas de nmap, Burp o Metasploit y deciden el siguiente paso.



Generación de payloads

Adaptan exploits conocidos a la versión y configuración específica del objetivo.



Orquestación de herramientas

Un agente central coordina múltiples utilidades de seguridad sin guion fijo.

¿Por qué IA en pentesting?



Razonamiento contextual

Los LLM interpretan resultados de escaneos y sugieren el siguiente paso



Velocidad y escala

Procesan miles de endpoints y CVEs en minutos, priorizando lo explotable.



Automatización de tareas repetitivas

Fuzzing, generación de payloads y correlación de hallazgos sin intervención constante.



Aumenta, no reemplaza

Libera al pentester humano de tareas repetitivas que conlleva muchas horas de análisis.

Roadmap de adopción en tu organización

1

Piloto controlado

IA solo en reconocimiento, sobre un activo no crítico

2

Métricas y validación

Compara hallazgos IA vs. manual, mide falsos positivos

3

Ampliar con supervisión

Escaneo y priorización con aprobación humana por paso

4

Integración continua

IA en el pipeline DevSecOps, con auditoría permanente

Enfoque tradicional vs. con IA

Enfoque tradicional

- Reconocimiento y enumeración manuales, dependientes de la experiencia del pentester
- Cobertura limitada por la cantidad de pentesters disponibles frente a los activos por asegurar
- Trabajo repetitivo: escaneo, fuzzing y correlación de hallazgos hechos a mano

Potenciado con IA

- Reconocimiento y priorización asistidos por modelos que cruzan señales en paralelo
- Automatiza fuzzing, generación de payloads y correlación de hallazgos
- Reduce el tiempo y el costo por prueba frente al enfoque manual



 EN CIFRAS

Lo que dicen los estudios sobre IA en pentesting

67%

de los equipos de red team ya usa al menos una herramienta de IA (vs. 18% en 2023)

40%

de las pruebas de penetración en grandes empresas incorporará IA para 2027

35%

menos tiempo en generar el reporte en pruebas de alcance medio

Fuente: encuesta SANS Institute (2025) y pronóstico de Gartner (2025), citados en Redfox Cybersecurity, "The State of AI Pentesting in 2026"; y Bishop Fox (tiempo de reporte).

 DATOS

Herramientas y frameworks reales de IA para pentesting

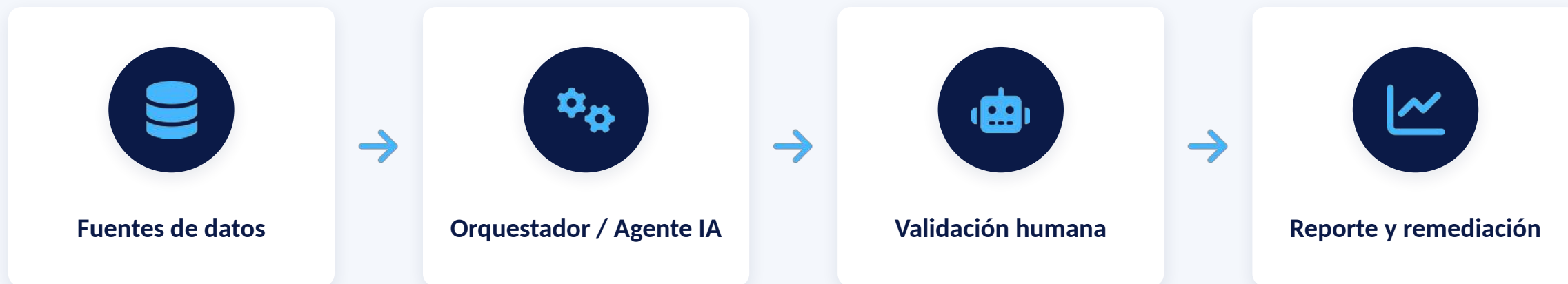
Herramienta	Enfoque de IA	Fase / uso principal	Madurez de IA
CAI (aliasrobotics)	Framework de investigacion "Cybersecurity AI": agentes especializados con benchmarks publicos (arXiv)	Todas las fases del pentest	Muy alta — validada academicamente (156x menor costo, hasta 3.600x mas rapido)
Strix (usestrix)	Multiagente "Graph of Agents": explota y valida con PoC real; SaaS + CI/CD	Reconocimiento, explotacion, reporte	Alta — produccion, auto-fix y validacion automatica
PentAGI (vxcontrol)	Orquestador, Researcher, Developer, Executor y Pentester con memoria vectorial	Todas las fases, con memoria persistente	Alta — orquestacion avanzada, 20+ herramientas
Shannon (KeygraphHQ)	Pentester de caja blanca: analiza codigo y ejecuta exploits reales antes de reportar	Reconocimiento y explotacion (XSS, SSRF, auth)	Media-alta — autonomo, acotado a Claude y 5 clases de vuln.
HexStrike AI (0x4m4)	Servidor MCP: expone 150+ herramientas ofensivas a agentes externos (Claude, GPT, Copilot)	Escaneo y explotacion (via agente externo)	Media — capa de herramientas; el razonamiento lo aporta el agente externo



IA orientada a Web3 y smart contracts

Herramienta	Enfoque de IA	Fase / uso principal	Madurez de IA	Costo
Slither-MCP (Trail of Bits)	Servidor MCP que conecta LLMs al motor de análisis estático de Slither	Auditoría y desarrollo: localizar funciones, callers/callees, correr detectores	Emergente — nov. 2025, sobre un motor (Slither) ya muy maduro	Gratuita (AGPLv3); licencia dual de pago para uso en SaaS
Solodit (Cyfrin)	Buscador + dataset de 50.000+ hallazgos históricos de auditorías	Reconocimiento previo: checklist y research de vulnerabilidades conocidas	Alta — referencia consolidada entre auditores web3	Gratuita (cuenta gratis; funciones extra con registro)
Hashlock AI Audit	IA propia entrenada en patrones de vulnerabilidad comunes	Escaneo automático pre-deployment de contratos	Beta — resultados en segundos, primera línea de defensa	100% gratuita
AuditSmart	10 agentes IA en paralelo (Groq LLaMA, Gemini, Slither y Claude)	Auditoría automatizada con reporte PDF y código de arreglo	Producción — motor de deduplicación y validación cruzada	Freemium (gratis limitado; planes de pago desde US\$19)
SmartContract Auditor.ai	Escaneo IA multi-lenguaje: Solidity, Vyper y Rust	Auditoría de vulnerabilidades y optimización de gas	Producción — adoptado por desarrolladores web3	Freemium (planes de pago)
Fethron AI Studio	Chat IA generalista con módulo de auditoría de Solidity	Chequeo rápido pre-auditoría: severidad y líneas afectadas	Emergente — herramienta secundaria de un estudio digital	100% gratuita, sin registro

Arquitectura de un pipeline pentest + IA



 CHECKLIST

Buenas prácticas para una adopción segura



Definir un alcance (scope) estricto y verificable antes de activar cualquier agente autónomo



Mantener un humano en el loop para acciones de explotación de alto impacto



Registrar y auditar cada decisión tomada por la IA (trazabilidad completa)



Validar hallazgos críticos manualmente antes de reportarlos al cliente



Actualizar y re-entrenar modelos con nuevas TTPs y CVEs de forma continua



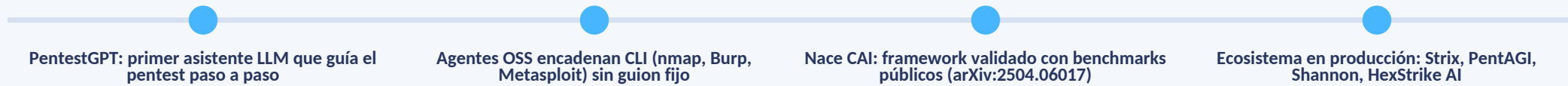
Establecer límites de scope y rate-limit técnico para cada agente autónomo



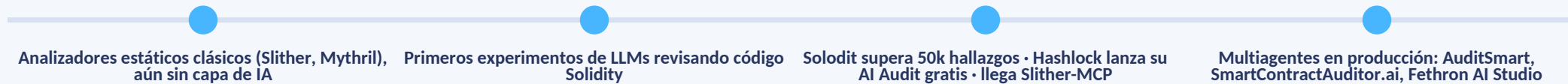


Evolución del pentesting con IA

PENTESTING GENERAL



IA APLICADA A WEB3 Y SMART CONTRACTS



PARA LLEVAR



1

La IA acelera, no reemplaza

Acelera reconocimiento, escaneo y priorización; el juicio del pentester sigue siendo clave.



2

Todo PoC se valida a mano

Un hallazgo o payload generado por un agente se confirma manualmente antes de reportarlo.



3

Empieza pequeño y con gobernanza

Piloto acotado, medición de falsos positivos y escalamiento con auditoría y humano en el loop.



Caso práctico con Hashlock

1

Elige el objetivo

Contratos vulnerables de práctica:
DeFiVulnLabs o Web3 (codeccs)

2

Sube el código a Hashlock

aiaudit.hashlock.com escanea el .sol con su
motor de IA

3

Revisa los hallazgos

Reentrancy, overflow y control de acceso,
con severidad y evidencia

4

Valida y documenta

Confirma cada hallazgo a mano antes de
reportarlo

Repositorios: github.com/SunWeb3Sec/DeFiVulnLabs · github.com/codeccs/Web3



¡Gracias!

¿Preguntas?

[linkedin.com/in/caamilacasas](https://www.linkedin.com/in/caamilacasas)