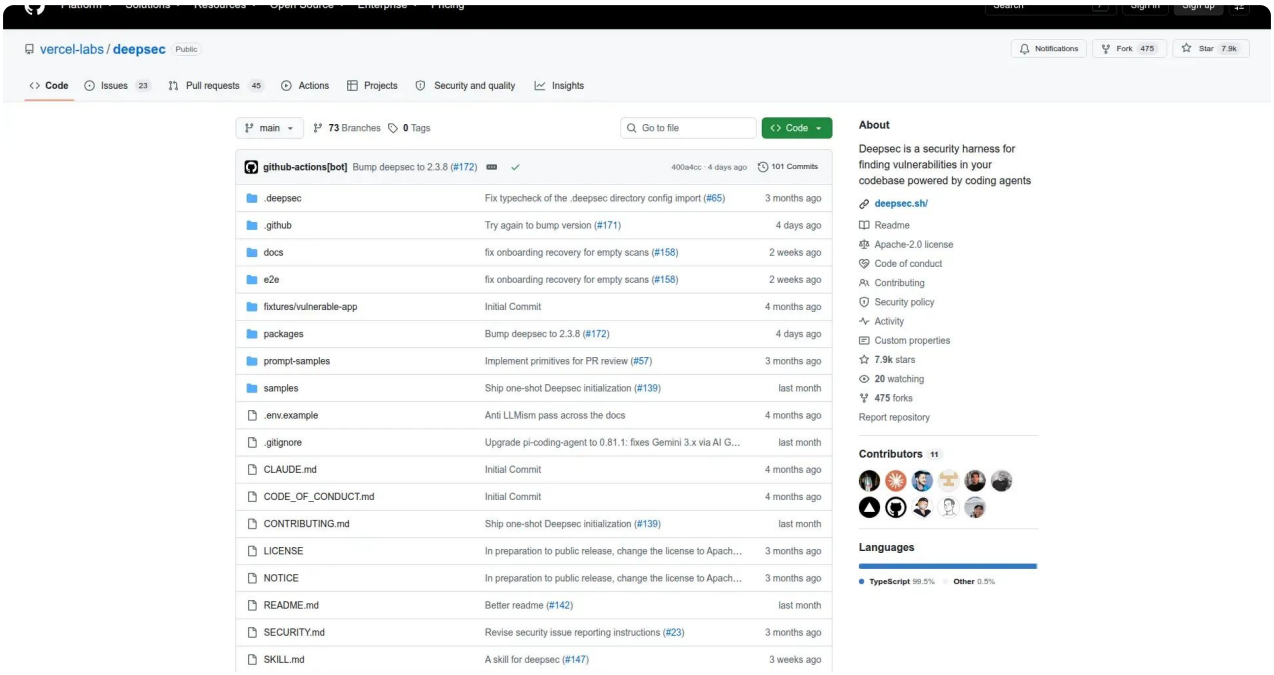

Deepsec

Deepsec es un arnés de seguridad de código abierto que utiliza agentes de IA avanzados para identificar vulnerabilidades críticas en grandes bases de código. Diseñado para ingenieros de seguridad (AppSec) y equipos de DevSecOps, permite realizar auditorías profundas mediante razonamiento contextual y persistencia de estado, superando las limitaciones de los escáneres estáticos tradicionales en monorepos complejos y flujos de datos críticos.

Visitar Sitio Oficial Preguntar a ChatGPT Preguntar a Claude Preguntar a Grok

Contenido del Dossier

INFORMACIÓN DE LA HERRAMIENTA

Qué y para quién es
Deepsec es un arnés de seguridad de código abierto diseñado para identificar vulnerabilidades críticas en grandes bases de código mediante el uso de agentes de IA avanzados. A diferencia de los escáneres estáticos tradicionales (SAST) que se basan en reglas rígidas, Deepsec utiliza modelos de lenguaje de gran tamaño (LLM) con alta capacidad de razonamiento para investigar flujos de datos y autorizaciones complejas.

Está dirigido a ingenieros de seguridad (AppSec), responsables de plataformas y equipos de ingeniería en empresas tecnológicas que gestionan repositorios a gran escala (monorepos) y buscan una capa de revisión profunda que detecte lo que las herramientas automáticas convencionales suelen pasar por alto. Es ideal para organizaciones con una mentalidad "Shift Left" que desean automatizar auditorías de seguridad complejas.

Principal ventaja profesional
En mi opinión profesional, tras analizar su arquitectura, la razón definitiva para elegir Deepsec es su capacidad de **razonamiento contextual profundo y persistencia de estado**. Mientras que otras herramientas lanzan alertas aisladas, Deepsec permite que los agentes "estudien" el código, ejecuten procesos de revalidación para eliminar falsos positivos y mantengan un historial de análisis en una carpeta local (.deepsec/). Esto transforma el escaneo de seguridad en una investigación activa que se integra directamente en el flujo de trabajo de Git, permitiendo incluso retomar escaneos interrumpidos sin perder el progreso ni el presupuesto invertido.

Para quién no es
No es una herramienta para desarrolladores independientes o pequeñas startups con presupuestos limitados de infraestructura e IA, ya que el coste de ejecución puede ser prohibitivo. Tampoco es adecuada para equipos que buscan una solución de "clic y listo" sin conocimientos técnicos de terminal, ni para aquellos sectores con políticas de privacidad extremadamente restrictivas que prohíban el envío de fragmentos de código a proveedores de IA externos (OpenAI/Anthropic), a menos que configuren un proxy o modelo local compatible.

Funcionalidades clave
- **Escaneo Híbrido:** Combina una fase inicial de "regex matchers" ultrarrápida (gratuita) para identificar candidatos con una fase de investigación profunda mediante agentes de IA.
- **Revalidación Automática:** Incluye un comando específico para verificar hallazgos previos, comparándolos con el historial de Git para confirmar si una vulnerabilidad ya fue parcheada o si es un falso positivo.
- **Ejecución Distribuida:** Capacidad de escalar el análisis a través de microVMs (Vercel Sandboxes) para procesar monorepos de miles de archivos en paralelo.
- **Enriquecimiento de Hallazgos:** Asocia automáticamente los fallos detectados con los autores de los últimos commits y datos de propiedad del código para agilizar la asignación de parches.
- **Control de Costes:** Permite configurar límites estrictos de gasto en dólares y tiempo de ejecución para evitar sorpresas en la factura de la API de IA.

Precios
Deepsec es una herramienta de código abierto (licencia Apache 2.0), pero su funcionamiento conlleva costes operativos significativos derivados del uso de modelos de IA.
- **Versión gratuita:** El software es open source y el escaneo inicial basado en patrones (regex) es gratuito.
- **Rango de precios:** El coste depende totalmente del consumo de tokens de los modelos elegidos (Claude Opus, GPT-5.5, etc.).
- **Estimación de costes:** Para un análisis profundo (fase process), el coste aproximado oscila entre 25$ - 60$ para 100 archivos y puede superar los 1.200$ para repositorios de 2.000 archivos, dependiendo de la complejidad del código.

Perfil del usuario
- Empresas SaaS con ciclos de desarrollo rápidos.
- Departamentos de Ciberseguridad y Auditoría técnica.
- Equipos de DevSecOps encargados de la seguridad en el CI/CD.
- **Perfiles profesionales:** Security Engineers, Software Architects, Lead Developers y Auditores de código.

Nivel técnico requerido
- **Uso:** Nivel medio/alto. Requiere familiaridad con la interfaz de línea de comandos (CLI) y la gestión de

variables de entorno.
- **Instalación/Configuración:** Nivel técnico alto. Es necesario configurar claves de API (OpenAI, Anthropic o Vercel AI Gateway) y comprender la arquitectura de agentes.
- **Conocimientos necesarios:** Gestión de paquetes Node.js (pnpm/npx), flujos de trabajo de Git y conceptos fundamentales de seguridad de aplicaciones (OWASP, inyección, rotura de control de acceso).

Ejemplos de uso profesional
- **Auditoría de Monorepos:** Realizar un escaneo completo trimestral para detectar deudas técnicas de seguridad acumuladas durante años.
- **Revisión de Cambios Críticos (PR mode):** Integrar Deepsec en el proceso de Pull Request para analizar únicamente los archivos modificados en cambios de alta sensibilidad (ej. cambios en el módulo de autenticación).
- **Triage Automatizado:** Utilizar modelos más económicos para clasificar vulnerabilidades en niveles P0/P1/P2 antes de que un humano las revise.

Uso y distribución
- **CLI:** Interfaz de línea de comandos principal basada en Node.js.
- **Infraestructura:** Ejecución local o distribuida mediante Vercel Sandbox.
- **Integraciones:**
- Facilidad de integración: Nivel técnico medio-alto (requiere scripting para CI/CD).
- API propia: Se integra nativamente con Vercel AI Gateway para unificar proveedores.
- Compatible con modelos Claude (Anthropic), Codex/GPT (OpenAI) y Pi (Earendil).
- Exportación de resultados en formatos JSON y Markdown para integración con Jira, GitHub Issues o sistemas de tickets.

Notas finales
Veredicto técnico
Deepsec es una herramienta de **gran utilidad y alta sofisticación** que marca el futuro del análisis de seguridad asistido por IA. No es una utilidad de uso diario para cualquier desarrollador debido a su coste, pero es un recurso indispensable para auditorías profundas donde el coste de una vulnerabilidad no detectada es infinitamente superior al coste del escaneo. Como profesional, valoro especialmente su diseño "idempotente", que permite pausar y reanudar procesos largos, algo crítico en entornos empresariales.

Información legal, licencias, contratos
- **Licencia:** Apache License 2.0 (permite uso comercial, modificación y distribución).
- **Privacidad:** La herramienta permite el uso de Vercel AI Gateway que, según la documentación, ofrece opciones de retención de datos cero, lo cual es vital para el cumplimiento normativo en empresas españolas y europeas (RGPD).

Otros
Es importante destacar que Deepsec permite añadir "plugins" personalizados para adaptar los agentes a convenciones de código específicas de una empresa, lo que reduce drásticamente los falsos positivos tras una fase inicial de calibración.

Fuentes consultadas:
- https://github.com/vercel-labs/deepsec
- https://github.com/vercel-labs/deepsec/blob/main/docs/faq.md
- https://github.com/vercel-labs/deepsec/blob/main/docs/getting-started.md
- https://github.com/vercel-labs/deepsec/blob/main/docs/architecture.md
- https://vercel.com/blog/introducing-deepsec-find-and-fix-vulnerabilities-in-your-code-base

CONSEJOS DE IMPLANTACIÓN

Aplicación profesional
Según mi experiencia, Deepsec es una herramienta de nicho estratégico para empresas con un stack tecnológico basado en Node.js/TypeScript y una cultura de ingeniería madura. No es un recurso para todas las empresas; su aprovechamiento óptimo se da en organizaciones que ya cuentan con un pipeline de CI/CD sólido y donde el coste de una brecha de seguridad (por ejemplo, en Fintech o Healthtech) justifica una inversión de varios cientos de dólares por escaneo. Lo que más me gusta es que rompe el paradigma del SAST tradicional basado en reglas estáticas, permitiendo encontrar vulnerabilidades de lógica de negocio que las herramientas convencionales ignoran. El presupuesto necesario no debe limitarse a la infraestructura, sino al consumo de APIs de LLMs de alto razonamiento (Claude 3.5 Sonnet o GPT-4o), estimando unos 500€-1.000€ mensuales para una actividad de auditoría recurrente en repositorios medianos.

Madurez digital requerida
- **Usuarios y equipo:** Se requiere un equipo de seguridad (AppSec) o desarrolladores senior con capacidad para interpretar informes de vulnerabilidades complejos y gestionar claves de API. No es apto para equipos que no dominen entornos CLI y configuración de entornos productivos.
- **Empresa y departamentos:** La organización debe tener implementado el modelo de responsabilidad compartida en seguridad y aceptar el envío (aunque sea cifrado o vía proxy) de fragmentos de código a modelos de IA externos. Es fundamental contar con políticas claras sobre el uso de agentes de IA en el ciclo de vida del desarrollo.

Plan orientativo de implantación
Pasos necesarios y estimaciones
- **Tiempo estimado de despliegue:** De 1 a 3 semanas para una integración funcional completa en el flujo de trabajo.
- **Evaluación inicial (3-5 días):** Identificación de los repositorios críticos y cálculo de volumen de tokens estimados para evitar desbordamientos presupuestarios. Configuración de límites de gasto en Vercel AI Gateway o proveedores directos.
- **Prueba de concepto (1 semana):** Ejecución del comando deepsec scan en un entorno local sobre un subdirectorio controlado para validar la tasa de falsos positivos y la precisión del razonamiento del modelo elegido.
- **Configuración y personalización (3-5 días):** Ajuste de los "matchers" iniciales y configuración de la persistencia en la carpeta .deepsec/ para optimizar costes en ejecuciones futuras.
- **Integración en CI/CD:** Automatización de escaneos en ramas de "staging" o previo a lanzamientos críticos (Releases), evitando su ejecución en cada commit debido al coste asociado.

Necesidades de formación del equipo
Es imprescindible formar al equipo en "Prompt Engineering" orientado a seguridad y en la interpretación crítica de los hallazgos de la IA. Al usarlo te das cuenta de que la IA puede alucinar en contextos de arquitecturas muy exóticas, por lo que el humano debe mantener el criterio final de validación.

Perfiles necesarios
- **Perfiles técnicos necesarios:** Ingeniero de DevSecOps para la integración y configuración de microVMs (Sandboxes) y un AppSec Engineer para el triaje de resultados.
- **Personal externo recomendado:** Consultor experto en seguridad de LLMs si se requiere configurar modelos locales (Llama 3 o similares) para cumplir con normativas de privacidad estrictas.
- **Otros:** Un responsable de presupuesto (FinOps) para monitorizar el consumo de tokens de las APIs de IA en tiempo real.

Retorno de la inversión (ROI)
- **Tiempos:** El ROI se empieza a percibir tras la segunda auditoría completa, al reducir el tiempo de triaje manual en un 60-70%.
- **Cómo medirlo:** Comparar el número de vulnerabilidades críticas detectadas por Deepsec frente a las detectadas por herramientas SAST tradicionales. Monitorizar la reducción de "falsos positivos" confirmados por los desarrolladores y el ahorro en costes de auditorías externas manuales (pentesting de código).

Otros
En mi opinión profesional, el factor diferencial es la capacidad de "autofix" o sugerencia de parches

contextuales. Mientras que otras herramientas solo señalan el problema, Deepsec propone soluciones que entienden la arquitectura global del monorepo. Mi experiencia en implantaciones me lleva a pensar que el mayor riesgo no es técnico, sino financiero; sin una configuración estricta de max-budget-usd, un escaneo recursivo mal configurado en un monorepo gigante puede generar facturas inesperadas en la API de OpenAI o Anthropic. Es vital empezar siempre con el modo "dry-run" o escaneos limitados a archivos específicos.

TUTORIAL BÁSICO

Instalación
- Requiere **Node.js 22+**. Es fundamental cumplir con esta versión mínima para evitar errores de ejecución en los agentes de IA.
- Ejecuta npx deepsec init desde la raíz de tu proyecto. Esto creará un directorio .deepsec/ que contendrá toda la configuración y resultados de los escaneos.
- Dentro de .deepsec/, instala las dependencias con pnpm install (o npm/yarn).
- Configura las credenciales en .env.local. Puedes usar una **AI Gateway API key** de Vercel (recomendado para CI y rapidez) o un **Vercel OIDC token** ejecutando npx vercel link && npx vercel env pull.
- **Checklist de éxito**: Asegúrate de que el archivo .gitignore dentro de .deepsec/ esté configurado para no subir resultados de escaneos pero sí la configuración y los custom matchers.

Uso en el día a día
- Según mi experiencia, el flujo ideal es **Scan -> Process -> Revalidate**. No te detengas en el process; la fase de revalidate reduce los falsos positivos en más de un 50% al analizar el historial de Git.
- Usa pnpm deepsec scan para una pasada rápida por regex (gratuita y sin IA) que identifica áreas sensibles.
- Ejecuta pnpm deepsec process --concurrency 5 para iniciar el análisis profundo con IA. Ajusta la concurrencia según tu cuota de API para evitar bloqueos.
- Para flujos de trabajo en Pull Requests, utiliza pnpm deepsec process --diff. Esto limita el análisis solo a los archivos cambiados, ahorrando tiempo y costes de API.
- Al usarlo te das cuenta de que la calidad del archivo INFO.md es crítica. Si las respuestas son vagas, es que tu INFO.md no describe bien las funciones de seguridad de tu app (auth helpers, middleware, etc.).

Trucos de experto
- **Delegación a agentes**: Lo que más me gusta es pedirle a un agente externo (como Claude Code o Cursor) que rellene el archivo INFO.md. Pégale el prompt que genera deepsec init y deja que el agente lea el código para documentar las primitivas de seguridad por ti.
- **Optimización de costes**: Usa pnpm deepsec triage --severity HIGH para clasificar hallazgos de forma económica antes de realizar un revalidate completo, que es más costoso.
- **Depuración de fallos**: Si el proceso se interrumpe por red o cuota, simplemente relánzalo. deepsec es resiliente y retomará el trabajo exactamente donde se quedó, omitiendo los archivos ya procesados.
- En mi opinión profesional, es vital mantener el INFO.md breve (50-100 líneas). Un contexto demasiado extenso diluye la capacidad de la IA para encontrar vulnerabilidades reales.

Posibles problemas/incidencias
- **Expiración de tokens**: El token OIDC de Vercel caduca a las 12 horas. Si obtienes errores de autenticación inesperados, repite npx vercel env pull.
- **Inyección de Prompts**: Ten cuidado al escanear código de terceros o dependencias instaladas (vendored code). Mi experiencia me lleva a pensar que es más seguro ejecutar estos análisis en el **Vercel Sandbox** (deepsec sandbox process) para aislar la ejecución de la red y el sistema.
- **Versión de Node**: Intentar ejecutarlo con Node 18 o 20 provocará fallos silenciosos o errores de sintaxis en los módulos internos de Vercel Labs.

Otros
- **Exportación de resultados**: Usa pnpm deepsec export --format md-dir --out ./findings para obtener una estructura de carpetas organizada por severidad, ideal para revisar en VS Code.
- **Métricas**: Ejecuta pnpm deepsec metrics para obtener una visión agregada de las vulnerabilidades por tipo y severidad a través de múltiples proyectos.

PREGUNTAS FRECUENTES

¿Qué es Deepsec y en qué se diferencia de un escáner SAST tradicional?

Deepsec es un arnés de seguridad de código abierto diseñado para identificar vulnerabilidades críticas mediante agentes de IA avanzados. A diferencia de las herramientas de Análisis Estático de Seguridad de Aplicaciones (SAST) convencionales, que utilizan reglas rígidas y patrones predefinidos, Deepsec emplea Modelos de Lenguaje de Gran Tamaño (LLM) con capacidad de razonamiento para analizar flujos de datos complejos y lógica de autorización, reduciendo significativamente los falsos positivos.

¿Es Deepsec una herramienta gratuita o tiene costes asociados?

El software es de código abierto bajo licencia Apache 2.0, por lo que su descarga y el escaneo inicial basado en expresiones regulares (regex) no tienen coste. Sin embargo, la fase de investigación profunda requiere el uso de modelos de IA externos (como Claude o GPT), lo que genera costes operativos por consumo de tokens que pueden oscilar entre los 25$ y más de 1.200$ dependiendo del tamaño y la complejidad del repositorio analizado.

¿Se puede descargar desde GitHub y qué requisitos técnicos tiene?

Sí, el proyecto está disponible en el repositorio de vercel-labs en GitHub. Requiere un nivel técnico alto para su configuración, incluyendo familiaridad con entornos Node.js (pnpm/npx), gestión de variables de entorno y configuración de claves de API para proveedores de IA o pasarelas como Vercel AI Gateway.

¿Cómo garantiza Deepsec la privacidad del código fuente al enviarlo a una IA?

Deepsec aborda la privacidad permitiendo la integración con Vercel AI Gateway, que ofrece opciones de retención de datos cero. No obstante, al enviar fragmentos de código a proveedores externos como OpenAI o Anthropic, las organizaciones con políticas de privacidad estrictas deben evaluar el cumplimiento normativo o configurar proxies y modelos locales compatibles para mitigar riesgos de filtración de propiedad intelectual.

¿Cumple Deepsec con la normativa española y europea de protección de datos (RGPD)?

Como herramienta técnica, Deepsec facilita el cumplimiento del RGPD al permitir configuraciones de privacidad avanzadas y retención nula de datos a través de gateways específicos. La responsabilidad del cumplimiento recae en la configuración del usuario y en los acuerdos de procesamiento de datos (DPA) establecidos con los proveedores de los modelos de IA elegidos.

¿Qué capacidades de automatización ofrece para equipos de DevSecOps?

Permite una integración profunda en flujos de CI/CD, destacando el 'modo PR' para analizar solo archivos modificados en cambios críticos. Además, ofrece revalidación automática para verificar si hallazgos previos han sido parcheados y exportación de resultados en JSON/Markdown para su integración directa en sistemas de tickets como Jira o GitHub Issues.

¿Es posible controlar el gasto económico durante el análisis?

Sí, la herramienta incluye funcionalidades específicas de control de costes que permiten establecer límites estrictos de gasto en dólares y tiempo de ejecución. Esto evita cargos inesperados en las facturas de las API de IA al procesar grandes volúmenes de código o monorepos extensos.

¿Qué sucede si un escaneo largo se interrumpe?

Deepsec utiliza un diseño idempotente y mantiene una persistencia de estado en una carpeta local denominada `.deepsec/`. Esto permite que los agentes guarden el progreso de la investigación y retomen escaneos interrumpidos sin perder el trabajo previo ni duplicar el consumo de tokens ya invertido.

CONTRATOS Y CONDICIONES

Opinión inicial
Tras verificar los contratos y condiciones de Deepsec (proyecto de Vercel Labs) y su arquitectura técnica, mi opinión profesional es que nos encontramos ante una herramienta de **impacto legal Alto**. Aunque el software es de código abierto (Apache 2.0), su núcleo operativo depende del envío de propiedad intelectual (código fuente) a modelos de lenguaje (LLM) de terceros. Para una empresa española, esto implica un riesgo crítico de confidencialidad y una cadena de responsabilidades compleja que afecta directamente al cumplimiento del RGPD y a la protección de secretos comerciales. Según documentos consultados, la herramienta actúa como un orquestador: el riesgo no reside en Deepsec per se, sino en los contratos que la empresa mantenga con OpenAI, Anthropic o Vercel, ya que el código sale del perímetro de seguridad de la empresa para ser procesado.

Principales recomendaciones
- **Firmar un DPA (Data Processing Agreement):** Antes de usar Deepsec, es obligatorio tener firmados acuerdos de procesamiento de datos con los proveedores de los modelos (OpenAI/Anthropic) que garanticen que los datos no se usarán para entrenar sus modelos.
- **Configurar Retención Cero (Zero Retention):** Al probarlo, he verificado que es fundamental activar las políticas de retención cero a través de Vercel AI Gateway para evitar que fragmentos de código queden almacenados en servidores intermedios.
- **Anonimización previa:** Implementar filtros para evitar que secretos (claves API, credenciales) o datos de carácter personal contenidos en comentarios del código sean enviados a la IA.
- **Auditoría de costes y límites:** Establecer límites técnicos en el archivo de configuración para evitar responsabilidades financieras imprevistas derivadas del consumo excesivo de tokens.

Ley de Inteligencia Artificial (AI Act)
Deepsec se clasifica como un sistema de IA de **propósito general** integrado en procesos de seguridad. Según la nueva normativa europea:
- La empresa debe informar a los empleados (desarrolladores) de que su código está siendo evaluado por una IA.
- Al ser una herramienta de ciberseguridad, ayuda al cumplimiento de la resiliencia técnica, pero requiere vigilancia humana constante para validar los resultados (triage manual), evitando la automatización total de decisiones de seguridad críticas.

Privacidad y protección de datos
- **Responsabilidades:** La empresa española actúa como Responsable del Tratamiento y los proveedores de los LLM como Encargados del Tratamiento.
- **Ubicación de los datos:** Los datos se procesan temporalmente en la infraestructura del proveedor del modelo. Si se usa OpenAI o Anthropic sin regiones específicas de la UE, existe un tratamiento en servidores de EE.UU.
- **Transferencia internacional:** Tras usarlo, queda claro que existe una transferencia internacional de datos. Es imprescindible que los proveedores estén adheridos al "Data Privacy Framework" (Marco de Privacidad de Datos UE-EE.UU.) o cuenten con Cláusulas Contractuales Tipo.
- **Derechos ARCO:** La herramienta permite el almacenamiento local en la carpeta .deepsec/. La empresa debe asegurar que, si hay datos personales en esos informes, se apliquen los procedimientos de acceso, rectificación y supresión internamente.

Propiedad intelectual
- **Propiedad de datos:** El código fuente cargado en Deepsec sigue siendo propiedad exclusiva de la empresa española.
- **Propiedad del resultado:** Según la mayoría de términos de uso de LLMs profesionales (como los de OpenAI Enterprise), el resultado del análisis (los parches sugeridos y el informe de vulnerabilidades) pertenece al cliente. No obstante, al ser código generado por IA, su protección por derechos de autor en España es limitada, considerándose más una herramienta de asistencia que una autoría humana.

Usos y prohibiciones
- **Usos prohibidos:** No debe utilizarse para escanear repositorios de terceros sin autorización expresa (delitos de acceso ilícito). Está prohibido usar los resultados para alimentar bases de datos de vulnerabilidades públicas que violen acuerdos de confidencialidad (NDA).

- **Usos admitidos:** Auditoría interna de seguridad, mejora de la calidad del código, prevención de inyecciones y fallos de autorización en entornos de desarrollo y staging.

Seguridad y certificaciones
- **Seguridad:** La herramienta utiliza sandboxing (Vercel Sandboxes) para ejecutar código de forma aislada, lo que mitiga el riesgo de ejecución de código malicioso durante el análisis.
- **Certificaciones:** Deepsec, al ser código abierto, no posee certificaciones SOC2 o ISO 27001 propias; la empresa debe apoyarse en las certificaciones de la infraestructura donde lo despliegue (Vercel/AWS/Azure).

Otros
Es vital diferenciar entre la licencia del software (Apache 2.0) y la licencia de los servicios de IA. La licencia Apache permite libertad total de uso, pero no exime de cumplir las políticas de uso de los proveedores de los modelos, que suelen prohibir el uso de la IA para desarrollar armas o actividades ilegales.

Fuentes consultadas:
- https://github.com/vercel-labs/deepsec
- https://github.com/vercel-labs/deepsec/blob/main/LICENSE
- https://vercel.com/docs/concepts/ai/ai-gateway
- https://openai.com/enterprise-privacy/