Hacking Asistido por IA

Automatizando Pentesting con VSCode, Copilot y MCP

arts·sec

Hugo Avila

Cybersecurity Consultant & Full Stack Developer

- Inicié mi recorrido en tecnología administrando sistemas e infraestructuras.
- Evolucioné hacia el Desarrollo Full-Stack, construyendo aplicaciones como ERPs (Enterprise Resource Planning) y herramientas de automatización.
- Actualmente desarrollo con un enfoque en **ciberseguridad**, especializándome en Pentesting Web y API en ArtsSec.com.

Linkedin: devhugoavila **X @hugok2k** **www.hugoavila.dev**

arts·sec

¿Es tarde para empezar en ciberseguridad?

NO, nunca es tarde.

- La ciberseguridad es multidisciplinaria → No necesitas ser experto desde el principio.
- Lo importante es aprender, practicar y tener curiosidad.

arts·sec

¿Cómo enfrentamos la complejidad actual?

- **Complejidad**: Aplicaciones web con microservicios, APIs, frameworks de JavaScript...la superficie de ataque es enorme.
- **Velocidad**: Los equipos de desarrollo (DevOps) despliegan código a una velocidad increíble. La seguridad necesita adaptarse (DevSecOps).
- **Repetición**: Muchas tareas iniciales son metódicas y repetitivas (reconocimiento, escaneos básicos, etc.).

¿Y si pudieras delegar lo monótono a un asistente digital?

arts·sec

Trabajo humano + IA

- **La IA se encarga de lo tedioso**: Automatiza las tareas repetitivas
- **El Humano se enfoca en lo complejo**: Usa su creatividad, intuición y experiencia para encontrar vulnerabilidades que una máquina pasaría por alto.

arts·sec

Introducción a Model Context Protocol (MCP)

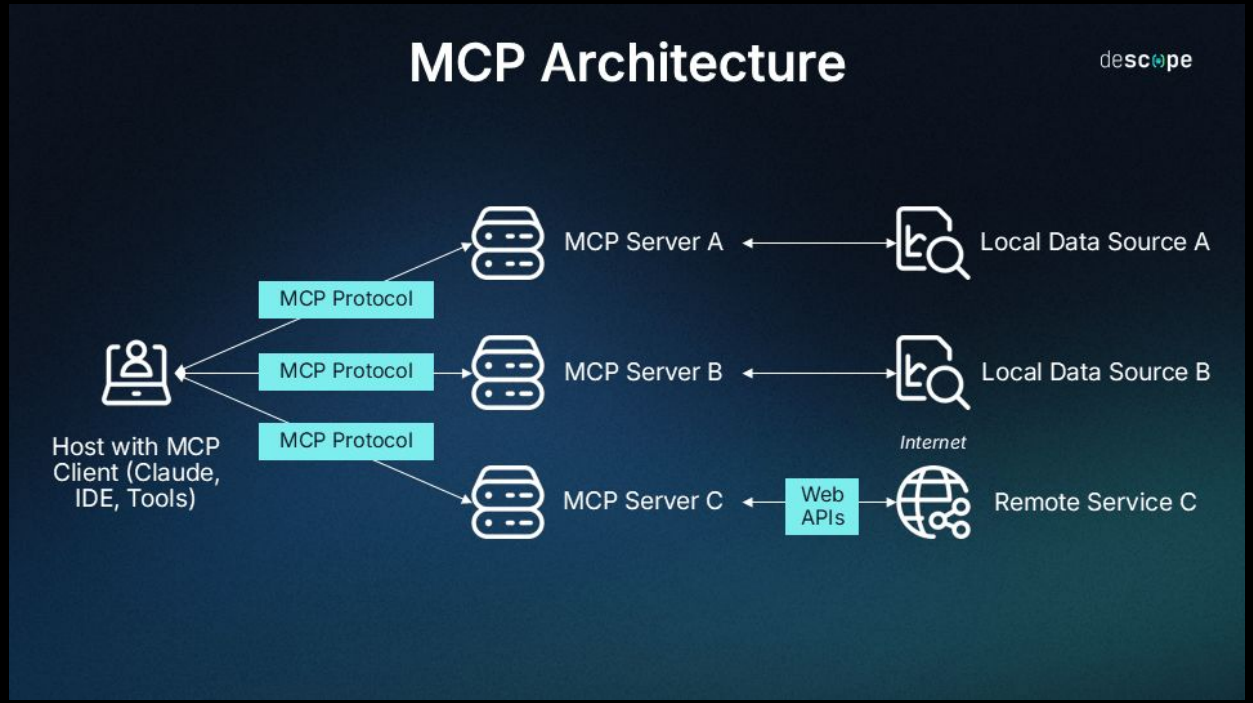
MCP (Model Context Protocol) diseñado por **Anthropic** y lanzado en noviembre de 2024 es un protocolo que actúa como un **puente** entre los grandes modelos de lenguaje (LLMs) como GitHub Copilot y tus herramientas locales. En lugar de que Copilot solo conozca el contexto actual del proyecto, MCP le permite **interactuar con otros programas y datos** en tu equipo.

Por defecto, Copilot vive dentro de VSCode y solo conoce su código. Con MCP, le damos 'ojos y oídos' para que pueda hablar con otros programas en su máquina, como Burp Suite, Playwright o incluso la terminal.

ANTHROP\C

arts·sec

¿Cómo funciona?



arts·sec

Nuestros aliados en esta misión

`@playwright`: navega por la web como un usuario. Lo usaremos para el reconocimiento inicial.

`@burp`: inspecciona y manipula todo el tráfico entre nuestro navegador y el objetivo. Ideal para ataques precisos.

`@filesystem`: puede leer archivos de configuración y escribir los reportes de vulnerabilidades.

Objetivo del taller

- Realizar un pentest básico a un sitio web de prueba.
- Le daremos a Copilot un "manual de instrucciones".
- Generar un informe.
- Todo dentro de VSCode,
- Sin salir del editor.
- Todo documentado paso a paso.

arts·sec

Paso a paso: Nuestra metodología

- Leer el alcance.
- Hacer reconocimiento inicial con herramientas de terminal.
- Navegar la aplicación para entenderla `(@playwright)`.
- Buscar vulnerabilidades comunes `(@burp)`.
- Documentar los hallazgos en un `informe` usando una plantilla.

Este flujo es replicable en cualquier target real (siempre respetando políticas de seguridad).

arts·sec

Hacking ético: Siempre con autorización

- Solo debemos auditar sistemas con permiso explícito.
- Respetar siempre los términos de servicio.
- Nunca exponer datos sensibles ni compartir resultados sin consentimiento.

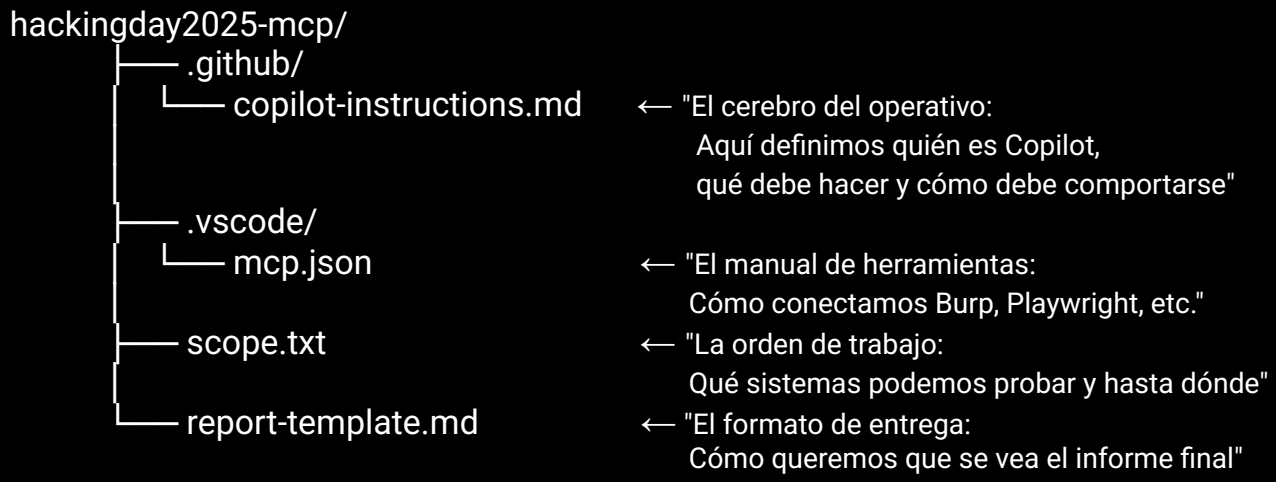
Hack the planet, but with ethics.

arts·sec

¡Agarramos la pala!

arts·sec

Estructura del proyecto



arts·sec

Lo que NO hace MCP

❌ NO reemplaza tu expertise en seguridad

❌ NO entiende contexto de negocio sin que se lo expliques

❌ NO toma decisiones éticas por ti

✅ SÍ automatiza tareas repetitivas

✅ SÍ acelera el proceso de documentación

✅ SÍ te permite enfocarte en el análisis profundo

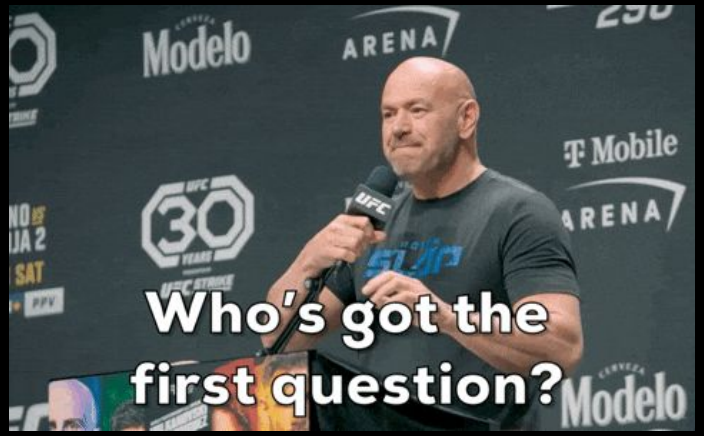
arts·sec

Referencias:

- **Repositorio MCP-Lab**: https://github.com/artssec/mcp_lab
- **MCP Burp**: https://github.com/PortSwigger/mcp-server
- **MCP Playwright**: https://github.com/microsoft/playwright-mcp/
- **Sitio oficial MCP**: https://modelcontextprotocol.io/
- **MCP Servers**: https://github.com/modelcontextprotocol/servers
- **MCP FileSystem**: https://github.com/modelcontextprotocol/servers/tree/main/src/filesystem
- **Customize AI VSCode**: https://code.visualstudio.com/docs/copilot/copilot-customization

arts·sec

¿Preguntas?



arts·sec

¡GRACIAS!

arts·sec