MCP Security for Enterprise Organizations

Claves para mejorar la postura seguridad en nuestros MCPs

EMPECEMOS →

Un poco sobre mí

- Trabajo en Cyber hace ya unos 7 años.
- Estoy cursando el Master de Seguridad Informática en la UBA.
- Co-fundador de ThreatX Security.
- Actualmente estoy como Sr Product Security y como Instructor de un Bootcamp.
- Tengo algunas certs de la industria.
- Soy el creador de PhiloCyber, mi proyecto personal.
- Amo los animales, sobre todos los perros.
- Me gustan los deportes, especialmente natación.
- Y disfruto mucho lo que hacemos.

PhiloCyber

Temas de hoy

01. Qué es MCP?
02. Arquitectura
03. Cómo Funciona?
04. Nivel de Adopción
05. Vulnerabilidades
06. Riesgos
07. Mitigaciones
08. Cierre y Conclusión

Milenios atrás...

2022

FUNDAMENTOS EXPERIMENTALES

- Function calling en LLMs (OpenAI)
- JSON-RPC 2.0 como estándar
- Arquitectura cliente-servidor

2024

CREACIÓN DE MCP

- En noviembre se comparte el protocolo, como un marco de trabajo open-source.

2021

SE FUNDA ANTHROPIC

- Hermanitos Amodei
- Ex VP de Investigación, OpenAI
- Ex VP de Políticas y Seguridad, OpenAI.

2023

MADURACIÓN Y AGENTES AUTÓNOMOS

- AI Agents autónomos.
- Integración con sistemas externos.
- Protocolos de comunicación LLM.

¿Qué es MCP?

Model Context Protocol

Es un protocol que define y estandariza como los LLMs se conectan con herramientas externas.

Qué busca?

Posicionado como el ***USB-C para aplicaciones de IA***, establece una arquitectura cliente-servidor que permite que los modelos de IA interactúen con diversas fuentes de datos y herramientas a través de una interfaz unificada.

¿Qué es un LLM?

"Es un tipo de inteligencia artificial entrenado con una gran cantidad de texto para comprender y generar lenguaje humano".

GPT-4

Qwen

Grok

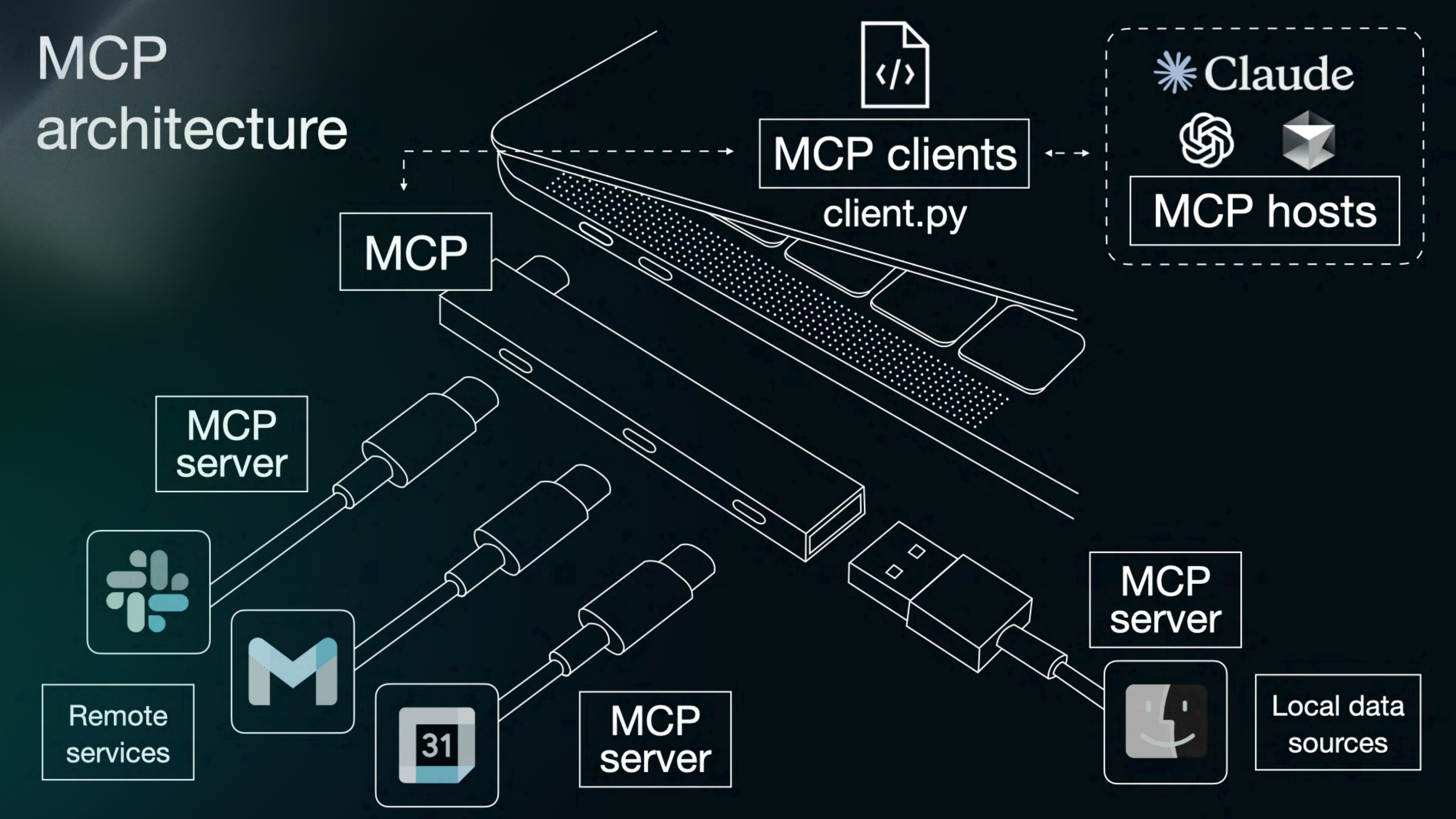
Mistral AI

Gemini

Falcon

DeepSeek-R1

Claude


MCP architecture
MCP clients
client.py
Claude
MCP hosts
MCP
MCP server
MCP server
MCP server
Remote services
Local data sources

Mecanismos de Transporte MCP

stdio (Local)

- Pipes STDIN/STDOUT
- Latencia µsegundos
- Solo local, seguro
- Uso: CLI, apps escritorio

Streamable HTTP (Remoto)

 Oficial

- Endpoint HTTP único
- 290-300 req/seg
- OAuth 2.1 + TLS
- Uso: Apps web producción

WebSocket (Remoto)

 Comunidad

- Full-duplex persistente
- Bidireccional tiempo real
- Implementación no oficial
- Uso: Apps tiempo real

Formato de Mensajes:

JSON-RPC 2.0 (todos los transportes)

Request
(con ID)

Notification
(sin ID)

Response
(ID coincidente)

MCP Client

Invokes **Tools**

Queries for **Resources**

Interpolates **Prompts**

MCP Server

Exposes **Tools**

Exposes **Resources**

Exposes **Prompts**

Tools

Model-controlled

Functions invoked
by the model

- Retrieve / search
- Send a message
- Update DB records

Resources

Application-controlled

Data exposed
to the application

- Files
- Database Records
- API Responses

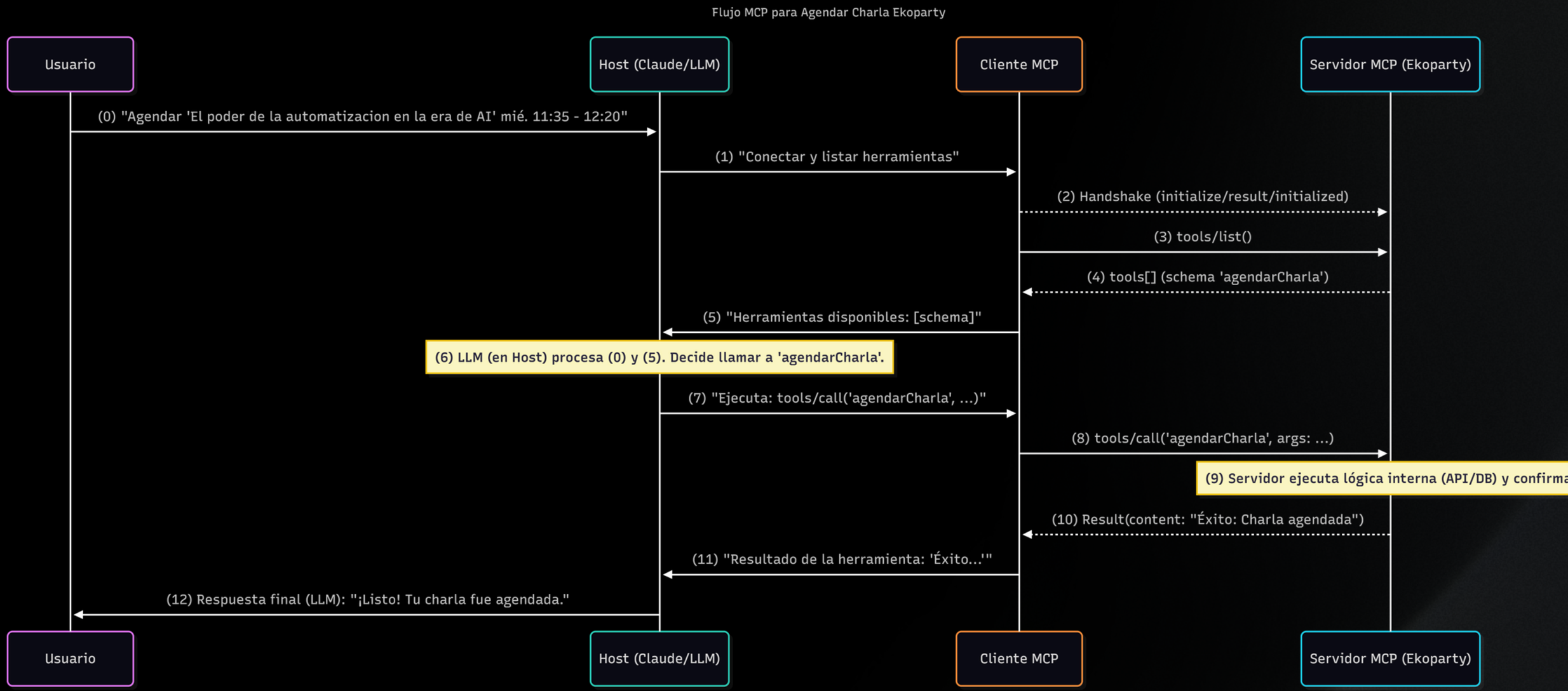
Prompts

User-controlled

Pre-defined templates
for AI interactions

- Document Q&A
- Transcript Summary
- Output as JSON

¿Cómo funciona?




MCP
SO HOT RIGHT NOW

Nivel de Adopción Actual

Adopción de IA

90% de los desarrolladores profesionales uitilizan IA en su día a día

Productividad

80% reporto que la IA incrementó su productividad

Dependencia

El 65% depende "moderadamente" de la IA en el trabajo

Calidad

59% afirmaron que la IA impactó positivamente en la calidad de código

Fuente: State of AI-assisted Software Development - DORA Research 2025

¿Cuál es el problema entonces?

API Calls >>> LLMs tooling

Introduce una nueva capa semántica, donde los Agentes de IA interpretan lenguaje natural para usar herramientas y recursos externos.


Y ASÍ QUEDAMOS...

Amenazas e Implicaciones

Tool Poisoning Attacks

Una inyección de prompts indirecta, donde el prompt malicioso está escondido en la descripción de la herramienta.

Tool Spoofing

Un atacante crea un servidor falso que parece legítimo, y los agentes AI se confunden, enviando datos confidenciales al servidor incorrecto.

The Rug Pull

Un servidor MCP opera bien y parece confiable, pero más tarde pasa a comportarse de forma maliciosa, atacando cuando menos lo esperan.

Prompt Injection

Alguien esconde instrucciones maliciosas dentro de la descripción o entrada para la herramienta... la IA las sigue y ejecuta acciones arbitrarias.

Unauthenticated access

No hay controles de identidad adecuados, seria como dejar la puerta abierta, cualquiera puede entrar y operar como si fuera legítimo.

Excessive Permissions

El servidor MCP tiene demasiados permisos o acceso a recursos. Si lo hackean, el daño puede ser enorme porque pueden acceder a todo.

¿Cómo mitigamos estos riesgos?

No hay secretos, ir siempre de lo más básico a lo más específico. Desde diseño y desarrollo seguro, hasta seguridad en profundidad, logs y monitoreo continuo y testeos.

ZERO TRUST

Nunca confiar en nada ni nadie, privilegios mínimos para el rol del usuario o la funcionalidad de la herramienta.

DEPENDENCIAS Y PARCHEOS

Revisar y monitorear dependencias constantemente,

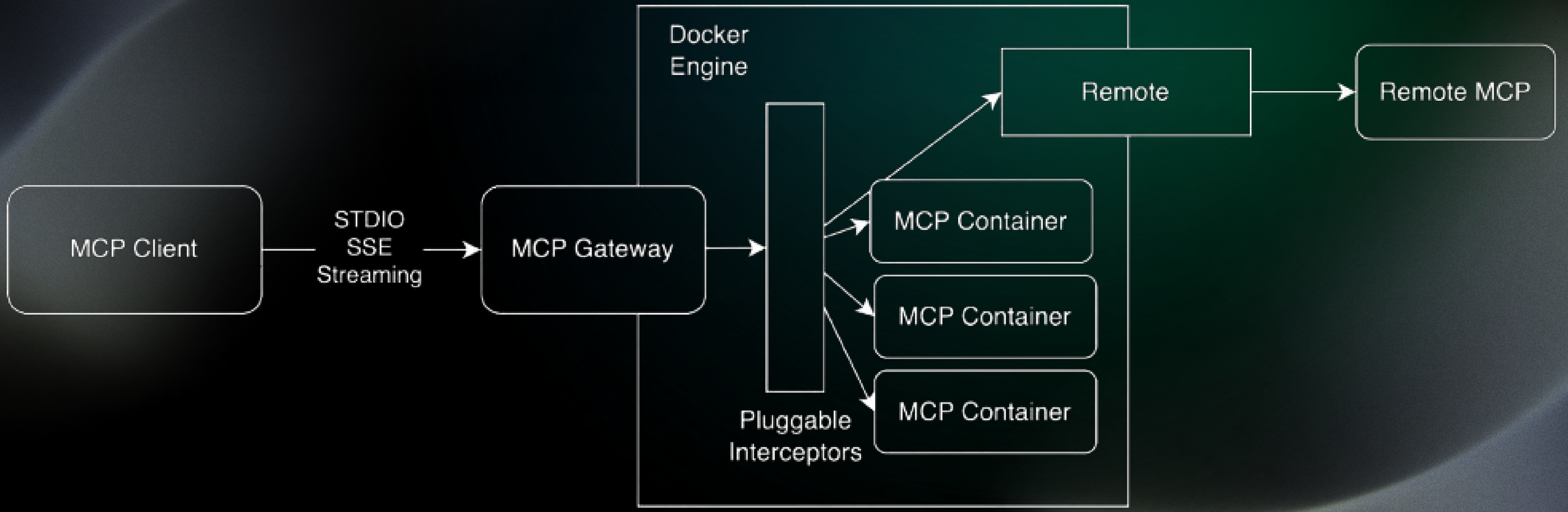
PLAN DE REPUESTA ANTE INCIDENTES

Armar un plan claro ante un eventual incidente, como aislar el sistema y controlar el impacto, roles y responsabilidades, monitoreo, y recuperación.

NO CONFIAR EN SERVIDORES DE TERCEROS

Siempre validar MCP servers, modelos, APIs de forma específica, escanear, testear, ver código y analizar si vale la pena tomar el riesgo intrínseco de usarlos.

MCP Gateway


Docker
Engine
Remote
Remote MCP
MCP Client
STDIO
SSE
Streaming
MCP Gateway
MCP Container
MCP Container
MCP Container
Pluggable
Interceptors

Conclusiones

1. EL PROTOCOLO REVOLUCIONÓ LA INTEGRACIÓN DE IA CON SISTEMAS EXTERNOS, PERO INTRODUJO NUEVOS RIESGOS DE SEGURIDAD.

2. LA ADOPCIÓN REQUIERE UNA ESTRATEGIA ZERO TRUST Y MONITOREO CONSTANTE.

3. LA VALIDACIÓN DE SERVIDORES MCP Y CONTROL DE PERMISOS SON CRÍTICOS PARA MITIGAR AMENAZAS.

Muchas gracias!

philocyber.com