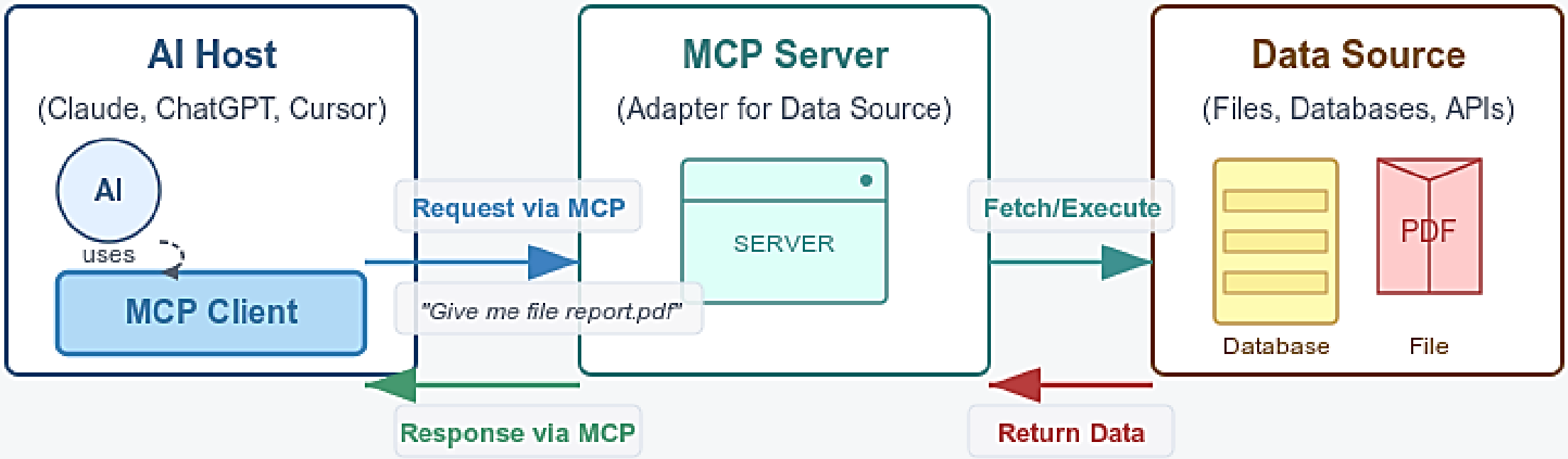
AI Hacking: armamento de la inteligencia artificial

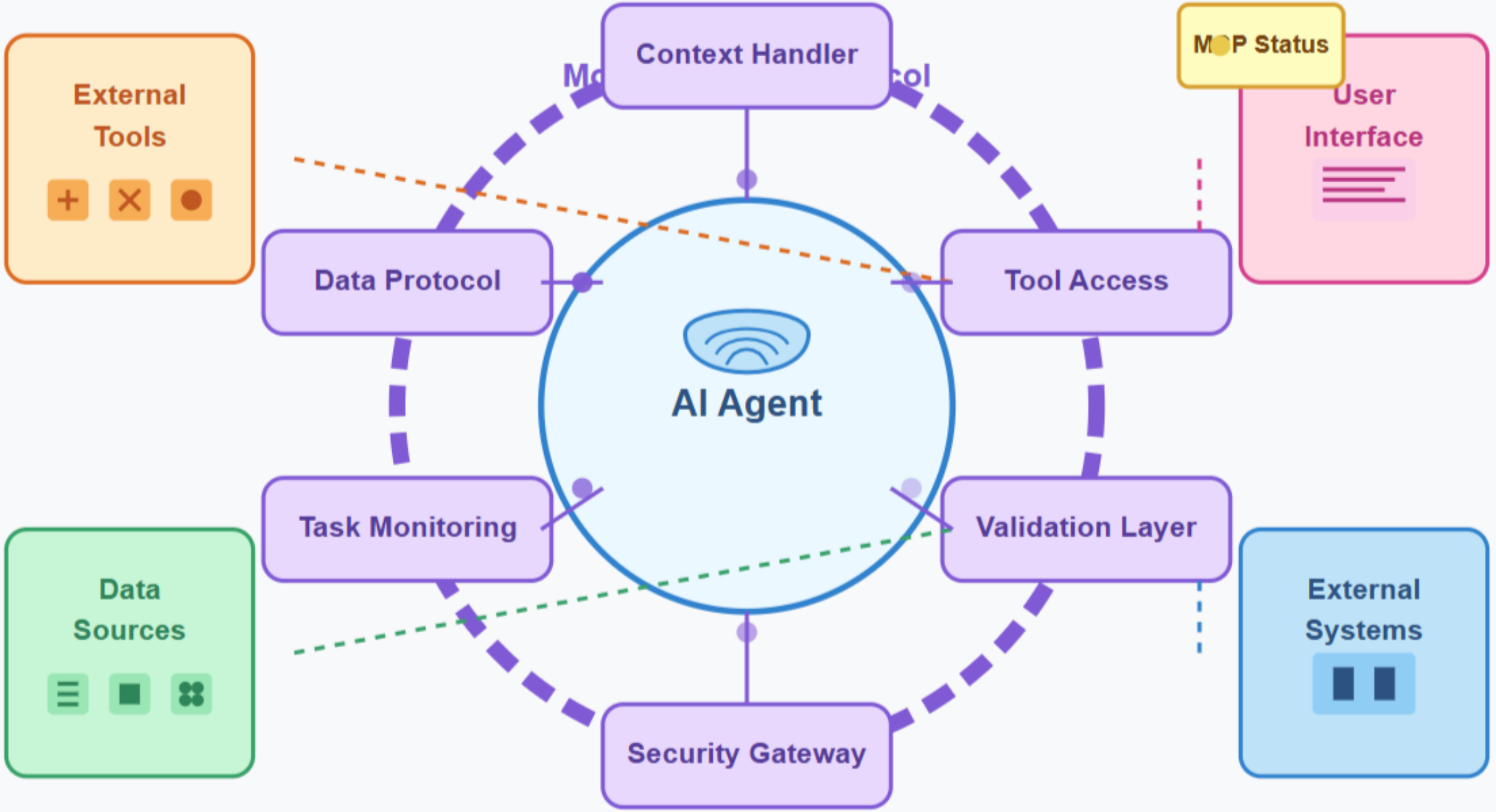
Con las capacidades de la inteligencia artificial potenciando las actividades de hacking, obtenemos una visión completa de todos los problemas, incluyendo las zonas periféricas para evaluar con precisión los posibles vectores de ataque, analizando la explotabilidad y el impacto en el negocio si la amenaza fuera explotada.

Model Context Protocol (MCP) Architecture

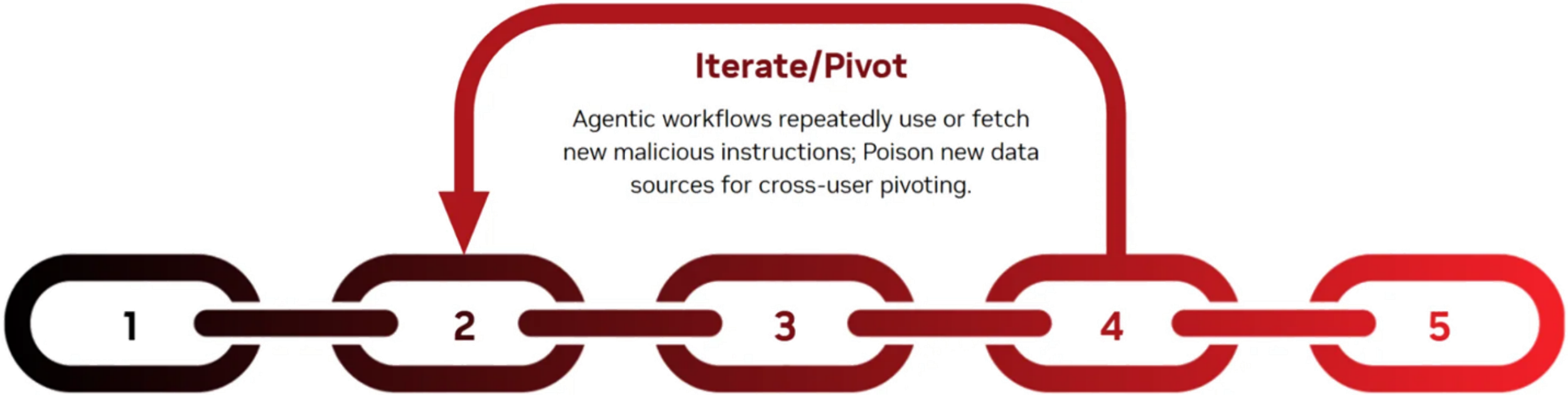


Model Context Protocol (MCP) Flow

The MCP Client translates AI requests into the standardized protocol format, communicates with MCP Servers, which then interact with external Data Sources.


External Tools
Context Handler
MCP Status
User Interface
Data Protocol
Tool Access
AI Agent
Task Monitoring
Validation Layer
Data Sources
External Systems
Security Gateway

IA Offensive Security killchain



Recon

Attacker maps system behavior to uncover weaknesses

Poison

Attacker places malicious data where the model can ingest it

Hijack

Malicious data hijacks model behavior, produces tainted outputs

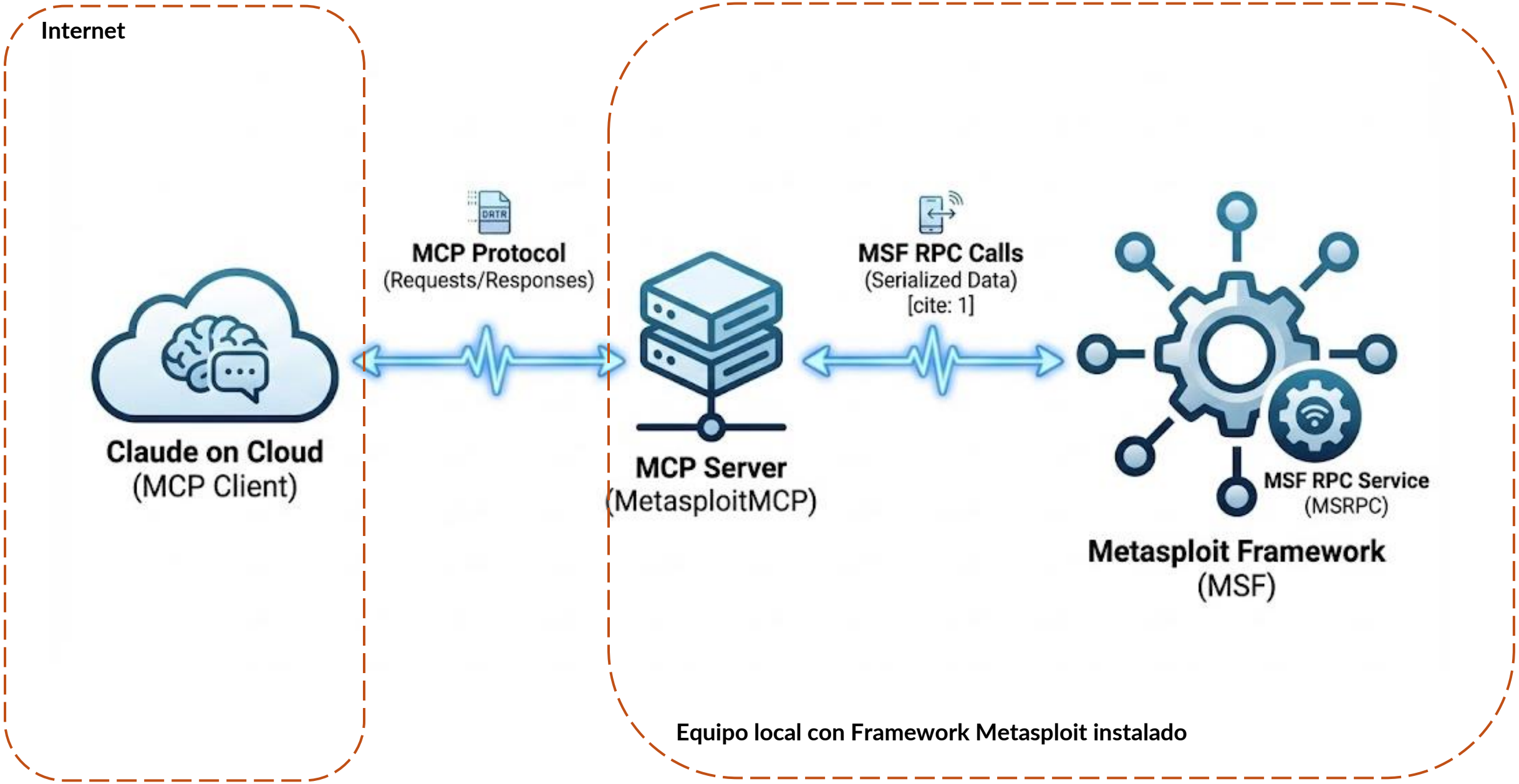
Persist

Exploit memory to maintain influence across sessions or in agentic loops

Impact

Trigger real-world actions from tainted outputs

Laboratorio & Herramientas


Internet
MCP Protocol
(Requests/Responses)
MSF RPC Calls
(Serialized Data)
[cite: 1]
Claude on Cloud
(MCP Client)
MCP Server
(MetasploitMCP)
MSF RPC Service
(MSRPC)
Metasploit Framework
(MSF)
Equipo local con Framework Metasploit instalado

Código

#Instalación

```
pip install -r requirements.txt

MSF_PASSWORD=yourpassword
MSF_SERVER=127.0.0.1
MSF_PORT=55553
MSF_SSL=false
PAYLOAD_SAVE_DIR=/path/to/save/payloads  # Optional:
```

Código

```
#Start the Metasploit RPC service:

msfrpcd -P yourpassword -S -a 127.0.0.1 -p 55553

# Run with HTTP/SSE transport (default)
python MetasploitMCP.py --transport http

# Run with STDIO transport
python MetasploitMCP.py --transport stdio
```

Recomendaciones para uso y aseguramiento

- Aprovechar los servidores existentes
- Crear servidores personalizados si la fuente de datos es propietaria o muy personalizada
- Aloja servidores adecuadamente: Para proyectos pequeños o desarrollo local, ejecutar servidores locales es lo más sencillo.
- Utilizar clientes de IA compatibles con MCP: Claude Desktop, Cursor IDE y frameworks como LangChain ofrecen compatibilidad con MCP.
- Probar e iterar