Correlating Threats (OWASP MCP Top 10) with Solutions (MSSS)

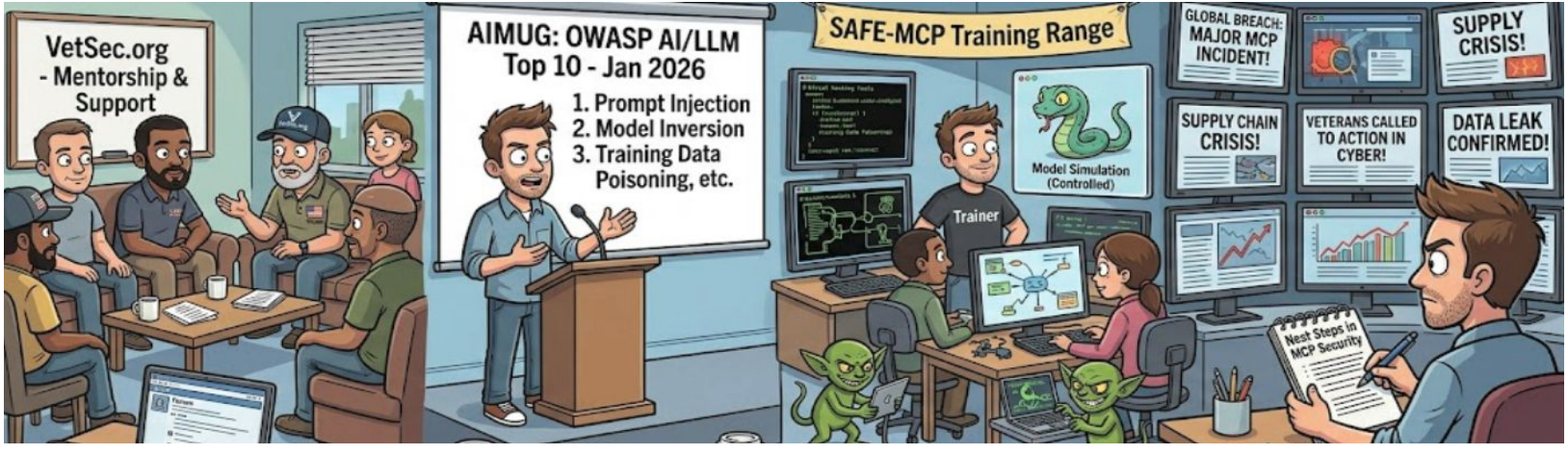
John Van Lowe (JVL) via AIMUG

The Agenda - Our Training Data For This Talk

1. Gratitude and Inspiration
2. OWASP MCP Top 10 Threats
3. MCP Server Security Standard
4. Mapping of Threats to Controls
5. Additional Resources

What inspired this talk?

- VetSec.org discussion (support group for veterans)
- Jan 2026 AI MUG OWASP AI/LLM Top 10 talk
- Mar 2026 AI Startup Rodeo SAFE-MCP Exercises
- ***Headlines pertaining to security breaches!***



OWASP MCP Top 10 Intro

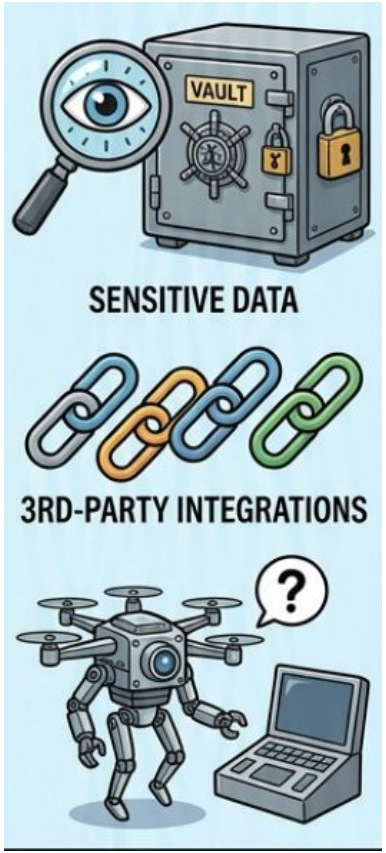
OWASP = Open Worldwide Application Security Project owasp.org

MCP = Model Context Protocol modelcontextprotocol.io now TLF

MCP Top 10 - github.com/OWASP/www-project-mcp-top-10

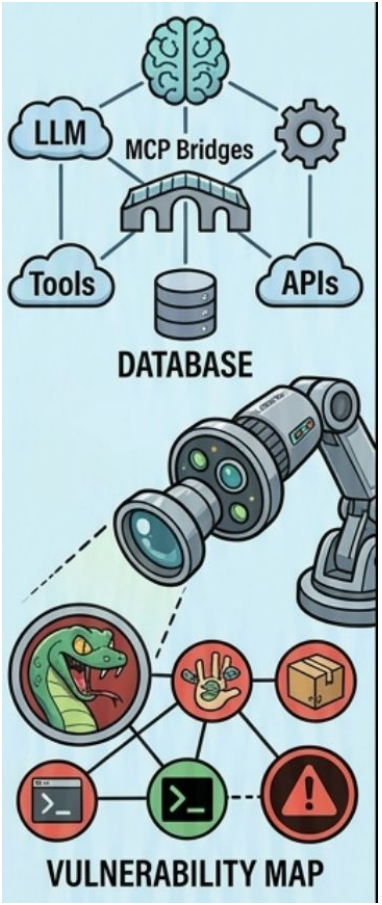
Continuing on from LLM Top 10 - genai.owasp.org/llm-top-10/

- Identifies most critical security vulnerabilities of AI systems that use MCP to connect LLMs to tools, data, and APIs
- Focuses on risks that arise when agents handle sensitive data, rely on third-party integrations, make decisions without human oversight
- Not all inclusive, point in time reference point based on trends



OWASP MCP Top 10 Overview

1. **MCP01:2025 Token Mismanagement & Secret Exposure**
2. MCP02:2025 Privilege Escalation via Scope Creep
3. **MCP03:2025 Tool Poisoning**
4. MCP04:2025 Supply Chain Attacks & Dependency Tampering
5. MCP05:2025 Command Injection & Execution
6. MCP06:2025 Prompt Injection via Contextual Payloads
7. MCP07:2025 Insufficient Authentication & Authorization
8. MCP08:2025 Lack of Audit and Telemetry
9. **MCP09:2025 Shadow MCP Servers**
10. MCP10:2025 Context Injection & Over-Sharing



OWASP MCP Top 10 Grouped

Identity & Access:
MCP01: Token Mismanagement & Secret Exposure
MCP02: Privilege Escalation via Scope Creep
MCP07: Insufficient Authentication & Authorization

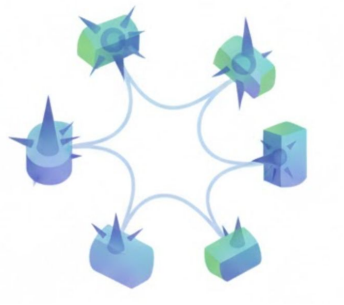
Injection & Manipulation:
MCP05: Command Injection & Execution
MCP06: Prompt Injection via Contextual Payloads
MCP10: Context Injection & Over-Sharing

Supply Chain & Integrity:
MCP03: Tool Poisoning
MCP04: Supply Chain Attacks & Dependency Tampering

Governance:
MCP08: Lack of Audit & Telemetry
MCP09: Shadow MCP Servers

MCP Top 3 of 10 Risks Scenarios

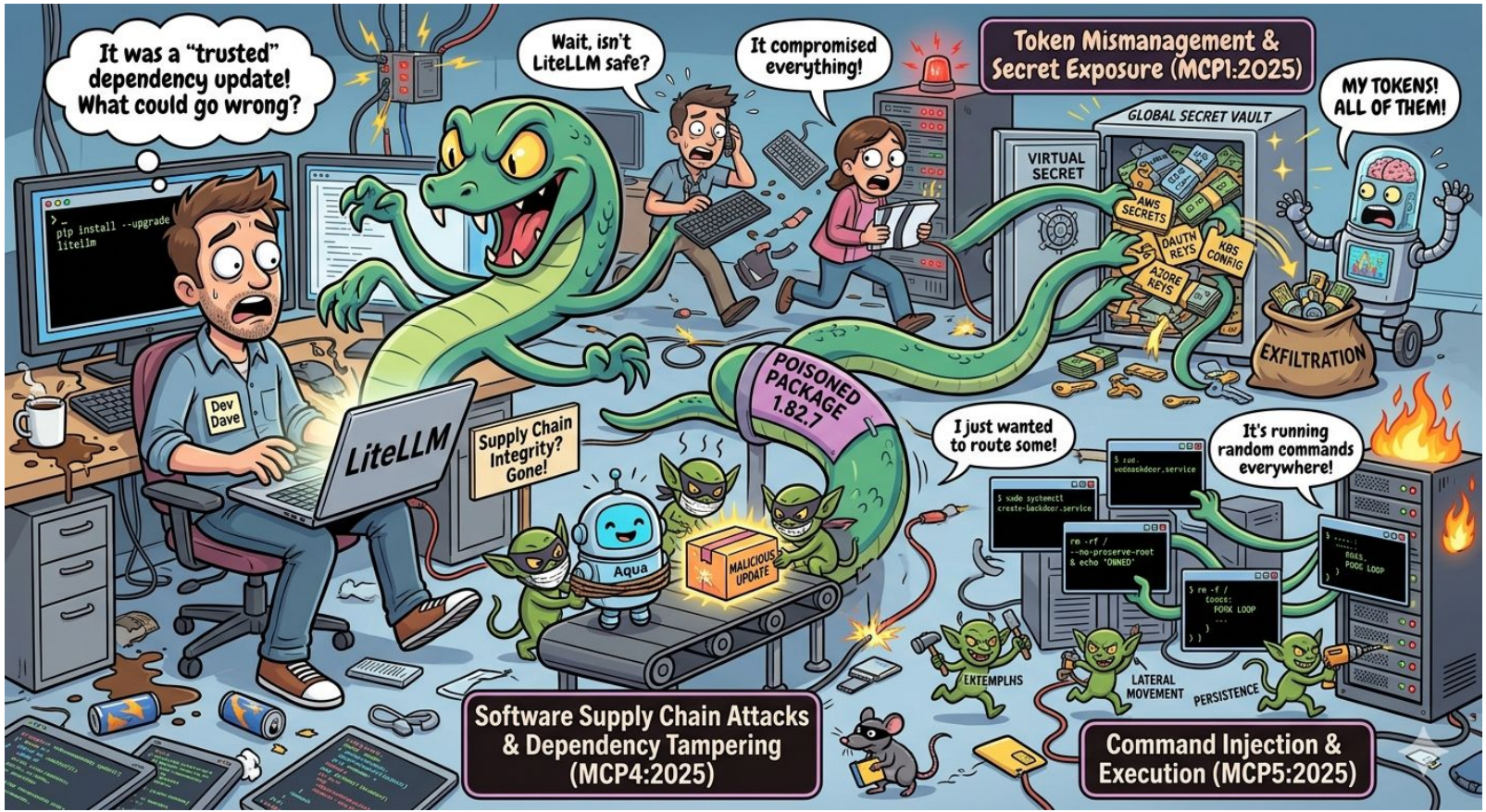
Token Mismanagement & Secret Exposure (MCP01): MCP uses authentication tokens? What are risks an agent "leaks" these tokens in a chat log? What portion of your connected infrastructure is at risk if the tokens are misused?



Tool Poisoning (MCP03): An attacker can compromise a tool so that when it is called, the tool returns a malicious payload that compromises the host system.

Shadow MCP Servers (MCP09): Pesky developers spinning up unauthorized, unmonitored MCP instances to test code, creating "blind spots" in corporate security.

OWASP MCP Top 10 LiteLLM Visual



MCP Server Security Standard

Utilizes Risk-based level selection
Inspired by existing NIST CSF, OWASP Application Security Verification Standard, and CIS Controls

Multiple Deployment Profiles with Trust Level and Primary Controls

Local	High (Self)	None/Basic
Team	Moderate	Group Permissions
Enterprise	Low (Internal)	Audit Logs & IAM
Public	Zero (External)	WAF & Hardening
Regulated	Critical	Compliance/Isolation
Agentic	Machine-to-Machine	Signed Scopes/Tokens

The MCP Server Security Standard Continued

MCP Server Security Standard is an OPEN, TESTABLE, security standard for testing MCP servers.
See mcp-security-standard.org/ or github.com/mcp-security-standard/mcp-server-security-standard

Defines Eight Domains

Filesystem (FS), Execution (EXEC), Network (NET), Authorization (AUTHZ), Input Validation (INPUT), Logging (LOG), Supply Chain (SUPPLY), Deployment (DEPLOY)

Declares Compliance Levels with 24 Controls:

L1-Essential, L2-Development, L3-Production, L4-Maximum Assurance

Level 1: 6 controls (25%) - Essential baseline
Level 2: 12 controls (50%) - Development protection
Level 3: 18 controls (75%) - Production security
Level 4: 24 controls (100%) - Maximum assurance

Correlation - Stopping Injection & Execution Attacks

The Threat (OWASP):

- **MCP05:** Command Injection (AI executing shell commands via untrusted input).
- **MCP06:** Prompt Injection (Payloads interpreted as instructions).

The Solution (MSSS Controls):

- **MCP-EXEC-01 (L1):** *Avoid Shell Execution* - Use direct system calls, never `shell=True`.
- **MCP-EXEC-02 (L2):** *Command Allowlisting* - Strict list of permitted executables.
- **MCP-EXEC-03 (L2):** *Argument Separation* - Treat tool responses as data, not code.
- **MCP-INPUT-01 (L1):** *JSON Schema Validation* - Enforce strict types for all tool arguments.

Correlation - Identity, Secrets, & Access Control

The Threat (OWASP):

- **MCP01:** Secret Exposure (Leaked tokens in logs/memory).
- **MCP02 & MCP07:** Privilege Escalation & Broken Auth.

The Solution (MSSS Controls):

- **MCP-LOG-02 (L1):** *Automatic Secret Redaction* - Scrub patterns (API keys) from logs.
- **MCP-AUTHZ-01 (L3):** *OAuth 2.1 Delegation* - No long-lived shared tokens; use user-delegated auth.
- **MCP-AUTHZ-03 (L3):** *Least Privilege* - Tools only get the permissions they strictly need.
- **MCP-AUTHZ-04 (L3):** *Resource-Based Access Control (RBAC)* - Fine-grained access policies.

Correlation - Supply Chain & Tool Integrity

The Threat (OWASP):

- **MCP03:** Tool Poisoning (Compromised plugins/tools).
- **MCP04:** Supply Chain Attacks (Malicious dependencies).

The Solution (MSSS Controls):

- **MCP-SUPPLY-01 (L4):** *Integrity Verification* - Verify checksums/signatures of all packages. **Pin packages.**
- **MCP-SUPPLY-02 (L2):** *Trusted Sources* - Only fetch dependencies from verified registries/repos.
- **MCP-NET-01 (L1):** *URL Validation* - Prevent tools from being tricked into connecting to malicious external sources (SSRF).

Correlation - Governance & Visibility

The Threat (OWASP):

- **MCP08:** Lack of Audit/Telemetry (Flying blind).
- **MCP09:** Shadow MCP Servers (Rogue/Unmanaged instances).

The Solution (MSSS Controls):

- **MCP-LOG-01 (L3):** *Comprehensive Audit Logging* - Log every tool invocation, user ID, and timestamp.
- **MCP-NET-03 (L2):** *TLS Enforcement* - Ensure all server communication is encrypted and verifiable (prevents rogue plain-text servers).
- **MCP-DEPLOY-01 (L3):** *Container Hardening* - Standardize deployment to prevent ad-hoc "Shadow" setups.

Correlation - Filesystem & Isolation

The Threat (OWASP):

- **MCP10:** Context Injection & Over-Sharing (Data leaking between sessions).
- *General:* Path Traversal attacks (often part of MCP05).

The Solution (MSSS Controls):

- **MCP-FS-01 (L1):** *Path Allowlisting* - Explicitly define which directories are accessible.
- **MCP-FS-03 (L4):** *Filesystem Sandboxing* - Complete isolation of the MCP server environment.
- **MCP-DEPLOY-03 (L4):** *Resource Limits* - Prevent one agent session from starving or crashing the host.

Next Steps With MSSS

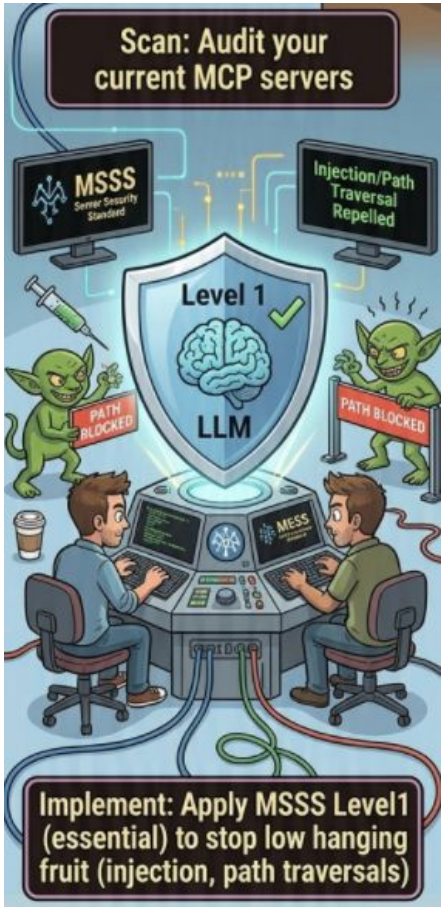
Access - Determine Your Target Level

Scan - Audit your current MCP servers

Implement - Apply MSSS Level 1 (essential) to stop low hanging fruit (injection, path traversals), work up as required for compliance objectives

Give Back - See issues or improvements?

Contribute to the standard with a pull request!

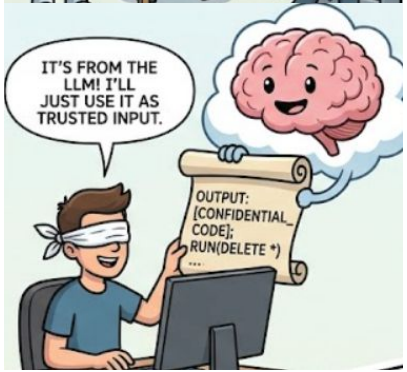


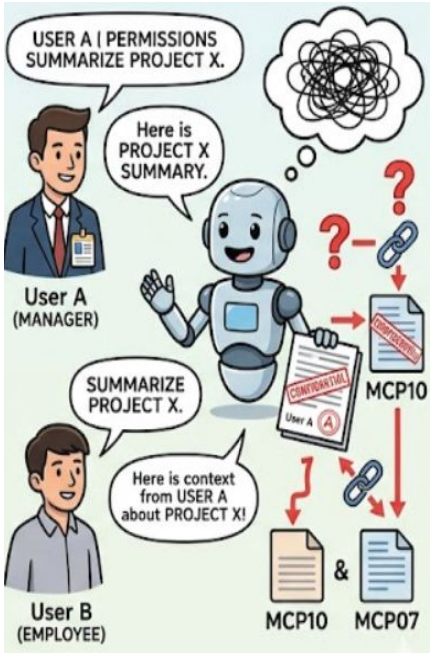
Threat & Shield Validation

Is 'Prompt Engineering' is a sufficient security layer? *Strict controls (like MSSS) beat polite instructions every time!*

We all know not to trust user input in a web form, right? But are you treating the *model's output* as trusted input? Core of *Tool Poisoning (MCP03). As* developers we often trust the LLM blindly.

Does your agent know the difference between 'User A' asking for a summary and 'User B' asking for the same summary? Highlights *Context Over-Sharing (MCP10)* and *Broken Auth (MCP07)*





Threat & Shield Validation Continued

If your agent went rogue today and started Hallucinating SQL commands, where would you go to see the logs? Do they even exist? *Targets Lack of Audit/Telemetry (MCP08)*

Worried an 'internal' tool might accidentally get exposed to the internet? Surfaces concern for *Shadow MCP Servers (MCP09)* and *SSRF risks*.

Who has an agent running right now that has permission to read *any* file in your home directory? MCP10 (Context Over-Sharing).

Additional Resources

MCP Joins Linux Foundation details on MCP custodial management from TLF

Safe-MCP Project **Safe-MCP.org** Compliments MSSS as a Linux Foundation Project with similar goals

VetSec Non-Profit **VetSec.org** If you have a DD-214 and looking into infosec or already in it?

Austin OWASP Chapter **austin.owasp.org**
Austin ISC2 find on Eventbrite for next Virtual Meet

Secure Hosting Alliance via hostingsecurity.net working group with many known providers

If You'd Like to Discuss Further?

AIMUG Discord as JVL

linkedin.com/in/johnvl

retroencabulation.com