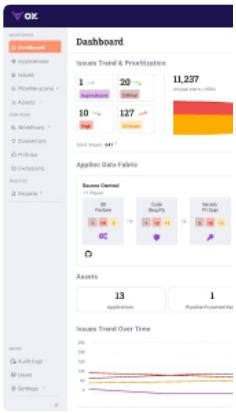
MCP Supply Chain Advisory: RCE Vulnerabilities Across the AI Ecosystem

April 15, 2026

Moshe Siman Tov Bustan (https://www.ox.security/author/moshe-siman-tov-bustan/)
Mustafa Naamnih (https://www.ox.security/author/mustafa-naamnih/)
Nir Zadok (https://www.ox.security/author/nir-zadok/)



Four separate exploits, deriving from one root cause in Anthropic's MCP SDK

OX Security researchers identified a systemic command injection vulnerability in Anthropic's MCP protocol that propagated across the AI ecosystem. This is the full disclosure advisory — including CVEs, affected platforms, and attack variants.

For more information:

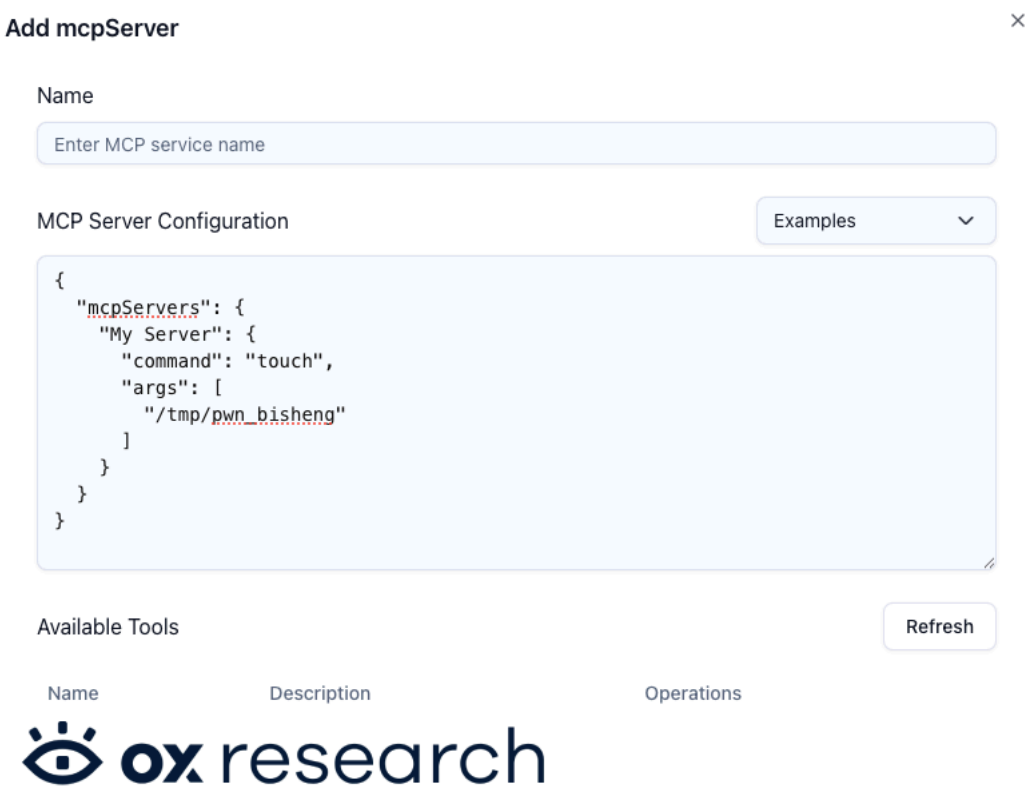
- Download the full eBook (https://www.ox.security/resource-category/whitepapers-and-reports/mother-of-all-ai-supply-chains/)for the complete findings
- Read the technical deep dive (https://www.ox.security/the-mother-of-all-ai-supply-chains-technical-deep-dive/)
- Read the main story (https://www.ox.security/blog/the-mother-of-all-ai-supply-chains-critical-systemic-vulnerability-at-the-core-of-the-mcp/)

MCP Vulnerability Family #1: Unauthenticated & Authenticated Command Injection via MCP STDIO

This family of vulnerabilities lets the attacker enter user-controlled commands which run directly on the server without authentication and without sanitization; any public server running with a publicly facing UI is vulnerable to this family of exploits.

In these scenarios, the attacker can simply identify the MCP adapter configuration logic which exposes this type of MCP configuration with STDIO commands, and enter a malformed JSON configuration with the arbitrary command.

These are some of the configuration screens which directly allow command execution:



CVE ID: Unassigned

Product: LangFlow

Link: https://github.com/langflow-ai/langflow (https://github.com/langflow-ai/langflow)

Description: LangFlow contains an unauthenticated remote command execution vulnerability within its MCP adapter configuration functionality. An attacker can obtain an authorization token via the publicly accessible /api/v1/auto_login endpoint and use it to add a malicious MCP server via an STDIO template. The application allows the user to supply arbitrary command and argument values, which are passed directly to StdioServerParameters (specifically within src/lfx/src/lfx/base/mcp/util.py) without adequate sanitization or an approved allowlist. Consequently, the underlying system executes the input as a subprocess, allowing an unauthenticated attacker to run arbitrary operating system commands and achieve remote command execution with the privileges of the LangFlow process.

Severity: Critical

Affected Versions: All versions of LangFlow.

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: CVE-2025-65720

Product: GPT Researcher

Link: https://github.com/assafelovic/gpt-researcher (https://github.com/assafelovic/gpt-researcher)

Description: When a victim accesses an attacker-controlled crafted HTML page, the page can trigger command execution and spawns a reverse shell on the machine running gpt-researcher locally.

This vulnerability can be exploited via entering the GPT-Researcher UI without authentication and entering a malicious MCP configuration which directly executes a command on the GPT-Researcher server.

Severity: Critical

Affected Versions: All versions of GPT Researcher.

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: CVE-2026-30623

Product: LiteLLM

Link: https://github.com/BerriAI/litellm (https://github.com/BerriAI/litellm)

Description: LiteLLM contains an authenticated remote command execution vulnerability in its MCP server creation functionality. The application allows users to add MCP servers via a JSON configuration specifying arbitrary command and args values. LiteLLM executes these values on the host without validation, enabling attackers to run arbitrary operating system commands. Successful exploitation may result in remote command execution with the privileges of the LiteLLM process.

Severity: Critical

Affected Versions: All versions of LiteLLM

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: TBD

Product: Undisclosed 1

Link: TBD

Description: A critical vulnerability in Undisclosed 1 allows remote attackers to execute commands directly from the UI's MCP configurations, by adding a new MCP server with a malicious payload.

Severity: Critical

Affected Versions: TBD

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: CVE-2026-30624

Product: Agent Zero

Link: https://github.com/agent0ai/agent-zero (https://github.com/agent0ai/agent-zero)

Description: Agent Zero 0.9.8 contains a remote command execution vulnerability in its External MCP Servers configuration feature. The application allows users to define MCP servers using a JSON configuration containing arbitrary command and args values. These values are executed by the application when the configuration is applied without sufficient validation or restriction. An attacker may supply a malicious MCP configuration to execute arbitrary operating system commands, potentially resulting in remote command execution with the privileges of the Agent Zero process.

Severity: Critical

Affected Versions: All versions of Agent Zero

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: Unassigned

Product: LangBot

Link: https://github.com/langbot-app/LangBot (https://github.com/langbot-app/LangBot)

Description: LangBot contains an authenticated remote command execution vulnerability within its MCP Server Configuration functionality. The application allows authenticated users to add an “STDIO” MCP server by specifying arbitrary command and argument values. Because the application utilizes StdioServerParameters to execute these given commands as a subprocess on the target machine without adequate validation or sanitization, an attacker can run arbitrary operating system commands. Successful exploitation may result in malicious actions such as data exfiltration or reverse shells, leading to remote command execution with the privileges of the LangBot process.

Severity: Critical

Affected Versions: All versions of LangBot

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: Unassigned

Product: Undisclosed 3

Link: Undisclosed 3

Description: Undisclosed 3 contains an authenticated remote command execution vulnerability within its MCP adapter configuration functionality. An attacker can exploit this by entering a malicious MCP configuration to run arbitrary operating system commands, potentially resulting in remote command execution with the privileges of the Undisclosed 3 process.

Severity: Critical

Affected Versions: All versions of Undisclosed 3

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: CVE-2026-30618

Product: Fay Digital Human Framework (Fay数字人)

Link: https://github.com/xszyou/Fay (https://github.com/xszyou/Fay)

Description: Fay (Fay数字人) contains an unauthenticated remote command execution vulnerability within its MCP adapter configuration functionality. The application allows users to add a new MCP server via the UI by supplying arbitrary command and argument values. Because the application processes this input through StdioServerParameters (specifically within faymcp/mcp_client.py) without proper sanitization or an approved allowlist, the underlying system directly executes the input as a subprocess. An attacker with access to the Web-GUI can exploit this by entering a malicious MCP configuration to run arbitrary operating system commands, resulting in remote code execution with the privileges of the Fay process.

Severity: Critical

Affected Versions: All versions of Fay Digital Human Framework (Fay数字人)

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: CVE-2026-33224

Product: Bisheng

Link: https://github.com/dataelement/bisheng (https://github.com/dataelement/bisheng)

Description: Bisheng contains an authenticated remote command execution vulnerability within its MCP tool configuration functionality. Because the platform allows open user registration, any attacker can easily gain the required access. Once authenticated, an attacker can navigate to the “Add MCP Server” interface and submit a crafted JSON configuration specifying an “stdio” MCP client type with arbitrary command and argument values. The application routes this input through src/backend/bisheng/mcp_manage/manager.py and passes it directly into StdioServerParameters (within

src/backend/bisheng/mcp_manage/clients/stdio.py) without sanitization or an approved allowlist. This results in the underlying system executing the malicious input as a subprocess, allowing the attacker to run arbitrary operating system commands and achieve remote code execution with the privileges of the Bisheng process.

Severity: Critical

Affected Versions: All versions of Bisheng <Will be updated>

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: CVE-2026-30616

Product: Jaaz

Link: https://github.com/11cafe/jaaz (https://github.com/11cafe/jaaz)

Description: Jaaz 1.0.30 contains a remote command execution vulnerability in its MCP STDIO command execution handling. A remote attacker can send crafted network requests to the network-accessible Jaaz application, where MCP is enabled, causing attacker-controlled commands to be executed on the server. Successful exploitation results in arbitrary command execution within the context of the Jaaz service, potentially allowing full compromise of the affected system.

Severity: Critical

Affected Versions: All versions of Jaaz, not affecting the official jaaz platform (https://jaaz.app/)

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: CVE-2026-30617

Product: Langchain-Chatchat

Link: https://github.com/chatchat-space/Langchain-Chatchat (https://github.com/chatchat-space/Langchain-Chatchat)

Description: LangChain-ChatChat 0.3.1 contains an unauthenticated remote command execution vulnerability in its MCP STDIO server configuration and execution handling. A remote attacker can access the publicly exposed MCP management interface and configure an MCP STDIO server with attacker-controlled commands and arguments. When the MCP server is started and MCP is enabled for agent execution, subsequent agent activity triggers execution of arbitrary commands on the server. Successful exploitation allows arbitrary command execution within the context of the LangChain-ChatChat service.

Severity: Critical

Affected Versions: All versions of LangChain-ChatChat (0.3.1)

Impact: Remote Command Execution (RCE), Full System Compromise

MCP Vulnerability Family #2: Unauthenticated command injection via direct STDIO configuration with hardening bypass

This family of vulnerabilities lets the attacker enter user-controlled commands which run directly on the server without authentication and without sanitization; but also with the ability to bypass already made protections and user input sanitization to the MCP configuration — both Upsonic and Flowise implemented protection from command injection by allowing only certain commands to run — such as “python”, “npm” and “npx”, removing the ability to directly send the command through the “command” parameter.

We were able to bypass this behavior by indirectly injecting the command via the allowed command’s arguments, for example — “npx -c <command>”.

CVE ID: CVE-2026-30625

Product: Upsonic

Link: https://github.com/Upsonic/Upsonic (https://github.com/Upsonic/Upsonic)

Description: Upsonic 0.71.6 contains a remote command execution vulnerability in its MCP server/task creation functionality. The application allows users to define MCP tasks with arbitrary command and args values. Although an allowlist exists, certain allowed commands (npm, npx) accept argument flags that enable execution of arbitrary OS commands. Maliciously crafted MCP tasks may lead to remote command execution with the privileges of the Upsonic process.

Upsonic decided to add a warning before using MCP tool initialization, warning users from running arbitrary commands through STDIO MCP configurations.

Severity: High

Affected Versions: All versions of Upsonic, a warning was issued on versions 0.72.0 and above

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: CVE-2026-40933

Product: Flowise

Link: https://github.com/FlowiseAI/Flowise (https://github.com/FlowiseAI/Flowise)

Description: Flowise contains a remote command execution vulnerability in its add MCP configuration inside the UI. The application allows users to define MCP tasks with arbitrary command and args values. Although an allowlist exists, certain allowed commands (npm, npx) accept argument flags that enable execution of arbitrary OS commands. Maliciously crafted MCP tasks may lead to remote command execution with the privileges of the Flowise server.

Severity: High

Affected Versions: >3.1.0

Impact: Remote Command Execution (RCE), Full System Compromise

MCP Vulnerability Family #3: Unauthenticated command injection via MCP configuration edit through prompt injection

In this vulnerability family, IDEs and coding assistants such as Windsurf, Claude Code, Cursor, Gemini-CLI and GitHub Copilot are vulnerable to command injection via their MCP JSON configuration, the only issued CVE is for Windsurf, as the user's prompt directly influences the MCP JSON configuration without user interaction.

All other IDEs and coding assistants we found this issue on — ask for at least one user interaction to allow the editing of the MCP JSON configuration file, even without the user knowing the contents and intent of the designated configuration, which in the IDE and coding assistant's standards does not qualify as a vulnerability as the user had to explicitly allow the file modification.

Windsurf Prompt Injection to Local RCE (CVE 2026 30615)

CVE ID: CVE-2026-30615

Product: Windsurf

Link: https://windsurf.com/ (https://windsurf.com/)

Description: A prompt injection vulnerability in Windsurf 1.9544.26 allows remote attackers to execute arbitrary commands on a victim system. When Windsurf processes attacker-controlled HTML content, malicious instructions can cause unauthorized modification of the local MCP configuration and automatic registration of a malicious MCP STDIO server, resulting in execution of arbitrary commands without further user interaction. Successful exploitation may allow attackers to execute commands on behalf of the user, persist malicious MCP configuration changes, and access sensitive information exposed through the application.

Severity: Critical

Affected Versions: All

Impact: Remote Command Execution (RCE), Full System Compromise

MCP Vulnerability Family #4: Unauthenticated command injection via network request, triggering hidden STDIO configurations

In this family of vulnerabilities, the insecure MCP STDIO configuration is not shown the the user in the server's Web-GUI, but the backend logic still contains STDIO processing logic, when an attacker crafts a malicious payload and sends it to the server, it triggers the STDIO configuration with an arbitrary command, triggering remote command execution on the server.

In these scenarios, the attacker enters either the production server of the project (For example — LettaAI, DocsGPT), and sees only an SSE or HTTP transport type configuration for the MCP server in the Web-GUI with no STDIO available. The attacker then configures one of these MCP servers and captures the network traffic using a local MITM proxy, then edits the request to change the sent transport type to be STDIO instead, and adds a "command" variable to the JSON's payload — which triggers the remote command execution.

Example of a Web-GUI configuration without an STDIO option —

Capturing the network request that contains only the HTTP MCP server configuration

Creating a new request with a modified JSON configuration, changing the “transport_type” to “stdio”, and adding a “command” and “args” variables to the JSON.

CVE ID: CVE-2026-26015

Product: DocsGPT (Formerly MemGPT)

Link: https://github.com/arc53/DocsGPT (https://github.com/arc53/DocsGPT), https://www.docsgpt.cloud/ (https://www.docsgpt.cloud/)

Description: DocsGPT contains a command injection vulnerability which affected both its production servers and local hosted services from its open source project.

An attacker crafting a network request for an MCP server configuration, and changing the transport type in the configured JSON to contain an STDIO type instead of SSE or HTTP, also adding an arbitrary command to the request's payload can achieve remote command execution.

Severity: Critical

Affected Versions: 0.15.0 (latest)

Note: This vulnerability was already patched on their official website.

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: Unassigned

Product: LettaAI

Link: https://github.com/letta-ai/letta (https://github.com/letta-ai/letta), https://letta.com/ (https://letta.com/)

Description: LettaAI contains a command injection vulnerability which affected both its production servers and local hosted services from its open source project.

An attacker crafting a network request for an MCP server configuration, with a JSON payload containing a server type of "stdio" alongside "command" and "arg" variables the request's payload can achieve remote command execution.

Severity: Critical

Affected Versions: All.

Note: This vulnerability was already patched on their official website.

Impact: Remote Command Execution (RCE), Full System Compromise

CVE ID: TBD

Product: Undisclosed 2

Link: TBD

Description: A critical vulnerability in Undisclosed 2 allows remote attackers to upload a specific ██████ which triggers remote commands a malicious MCP STDIO configuration.

Severity: Critical

Affected Versions: TBD

Impact: Remote Command Execution (RCE), Full System Compromise

Won't Be Patch Vulnerabilities: Our Rejected Disclosures

We disclosed the vulnerabilities to many other vendors and maintainers, this is a partial list of those we can publicly mention, that contain the MCP STDIO code that is able to execute code directly without sanitization, where vendors and maintainers disregarded our findings as expected behavior, either by stating one of the following

The system is designed to let the user execute code directly, and this is by design

Their code is the transport layer, and the developers using it are liable for their own implementation's security.

The code is being executed inside a sandbox environment (like Docker) which prevents abuse and the ability to reach other users' data.

While this statement is correct, it is still possible for threat actors to employ different attack scenarios like crypto mining, using their services as proxy, and getting free compute power.

Root & Transport Layer

Anthropic — Model Context Protocol

LangChain — langchain-mcp-adapters

FastMCP

browser-use/browser-use

Amazon, awslabs — run-model-context-protocol-servers-with-aws-lambda

NVIDIA — NeMo-Agent-Toolkit

IDEs & Coding Agents

Gemini-CLI

Claude Code

GitHub Copilot

Cursor

Applications

OpenHands

PromptFoo

Firebase Studio

Other STDIO MCP Vulnerabilities: Reported by other vendors and maintainers

After our research and disclosure process, we found that vulnerabilities based on this same core issue were reported by different researchers as single findings relating to untrusted MCP STDIO input during 2025-2026.

CVE-2025-49596 — MCP inspector

CVE-2026-22252 — LibreChat

CVE-2026-22688 — WeKnora

CVE-2025-54994 — @akoskm/create-mcp-server-stdio

CVE-2025-54136 — Cursor

Stay Updated

Sign-up to receive the latest news, exclusive updates, and product insights from OX

Business email*

Subscribe

Related Content

October 23, 2025

OX Research: AI Code Not Inherently Less Secure, but "Army... (https://www.ox.security/blog/ox-research-ai-code-not-inherently-less-secure-but-army-of-juniors-effect-undermines-software-security/)

By Eyal Paz

November 24, 2025

The Second Coming: Shai-Hulud Is Back at It — How... (https://www.ox.security/blog/the-second-coming-shai-hulud-is-back-at-it-how-to-protect-your-org/)

By Alexander Chailytko

November 27, 2025

OpenAI — Mixpanel 3rd Party Data Breach (https://www.ox.security/blog/openai-mixpanel-3rd-party-data-breach/)

By Moshe Siman Tov Bustan

It's time to secure your code the way software is built

Learn how VibeSec makes security part of your product creation from code to runtime