УДК 004.056.5:004.8:004.89

# ЗАСТОСУВАННЯ ZERO TRUST ARCHITECTURE ДЛЯ ЗАБЕЗПЕЧЕННЯ БЕЗПЕКИ AI-АГЕНТІВ НА БАЗІ MODEL CONTEXT PROTOCOL

Пестун А.М.
e-mail: andrii.pestun@nure.ua
Науковий керівник – к.т.н., проф. Шевченко О.Ю.
Харківський національний університет радіоелектроніки, каф. ШІ
м. Харків, Україна

This paper proposes an approach to integrating Zero-Trust Architecture (ZTA) principles, as defined by NIST SP 800-207, into the security layer of AI agents operating via the Model Context Protocol (MCP). The MCP specification enables LLM-based agents to invoke external tools through a standardized JSON-RPC 2.0 interface, but does not prescribe a comprehensive security policy. The proposed method adapts the Policy Engine, Policy Administrator, and Policy Enforcement Point triad to intercept and validate every MCP tool invocation. Microsegmentation is applied at the MCP server level, ensuring each server operates within an isolated trust zone with least-privilege access. Continuous authentication monitors agent behavior patterns to detect anomalous action sequences, enabling real-time revocation of permissions.

Стрімкий розвиток Large Language Models (LLM) та поява агентних архітектур на їх основі створюють нові можливості для автоматизації робочих процесів. Model Context Protocol (MCP), представлений компанією Anthropic у листопаді 2024 року, є відкритим стандартом підключення AI-систем до зовнішніх інструментів та даних через уніфікований J SON-RPC 2.0 інтерфейс [1]. Протокол реалізує трирівневу архітектуру Host – Client – Server із серверними примітивами Tools, Resources та Prompts, а також двома транспортними механізмами: stdio для локальних процесів та Streamable HTTP для віддаленого доступу. Масштаб прийняття протоколу є значним – понад 8 мільйонів завантажень SDK на тиждень станом на середину 2025 року [2], що актуалізує проблему забезпечення безпеки MCP-агентних систем.

Відмінність MCP від традиційних API полягає у недетермінованому характері потоку управління: мовна модель самостійно обирає послідовність викликів інструментів, що унеможливлює застосування класичних методів статичного контролю доступу. Аналіз 1899 відкритих MCP-серверів [2] підтвердив, що AI-керований потік створює принципово нові ризики для стійкості та безпеки. Hou X. та співавтори [3] побудували таксономію з 16 сценаріїв атак за 4 типами зловмисників, серед яких tool poisoning, context poisoning та ексфільтрація даних. He Y. та співавтори [4] показали, що AI-агенти з доступом до зовнішніх інструментів створюють унікальні безпекові виклики, включаючи prompt injection та несанкціоноване виконання операцій.

Для вирішення цієї проблеми запропоновано адаптацію Zero-Trust Architecture (ZTA), визначеної стандартом NIST SP 800-207 [5], до архітектури MCP-агентних систем. Стандарт базується на принципі "never trust, always verify" та визначає три логічні компоненти: Policy Engine (PE), Policy Administrator (PA) та Policy Enforcement Point (PEP). Запропонований підхід реалізує ці компоненти як проміжний безпековий шар між MCP-клієнтом та MCP-серверами.

Policy Enforcement Point перехоплює кожен JSON-RPC запит від AI-агента до MCP-сервера. На відміну від традиційних мережевих PEP, запропонований компонент аналізує семантичний контекст виклику: назву інструменту, структуру аргументів, відповідність поточному завданню користувача та послідовність попередніх дій агента. Policy Engine обчислює рівень довіри до кожного запиту за динамічною політикою, що враховує принцип мінімальних привілеїв та виявляє аномальні патерни поведінки. Policy Administrator генерує тимчасові session-scoped дозволи з обмеженням за часом та обсягом доступних операцій, забезпечуючи гранулярний контроль на рівні окремого виклику інструменту.

Мікросегментація реалізується на рівні MCP-серверів: кожен сервер функціонує як ізольований сегмент безпеки з індивідуальною політикою доступу. MCP-сервер файлової системи обмежується конкретними директоріями через механізм Roots протоколу, сервер бази даних – визначеними таблицями та типами запитів, сервер зовнішніх end-points – дозволеними адресами та типами запитів. Такий підхід гарантує, що компрометація одного сервера не надає зловмиснику доступу до ресурсів інших серверів, мінімізуючи радіус ураження.

Безперервна аутентифікація доповнює статичну верифікацію динамічним моніторингом поведінки агента протягом усього сеансу. Система відстежує послідовності викликів інструментів та порівнює їх з очікуваними шаблонами для відповідного типу завдань. Виявлення аномальної послідовності – наприклад, спроба читання конфіденційного файлу після серії нерелевантних запитів – ініціює повторну верифікацію або негайне відкликання дозволів. Усі операції журналюються для забезпечення повного аудиту дій AI-агента.

Практична реалізація запропонованого підходу передбачає впровадження безпекового шару як окремого проміжного компонента без модифікації специфікації MCP, забезпечуючи зворотну сумісність з існуючими MCP-серверами та клієнтами. Архітектурна схема запропонованого рішення наведена на рисунку 1. На відміну від мережевих проксі, компонент функціонує на рівні застосунку та має доступ до повного контексту взаємодії агента з інструментами, що є принциповою умовою для виявлення семантичних атак типу prompt injection та tool poisoning.

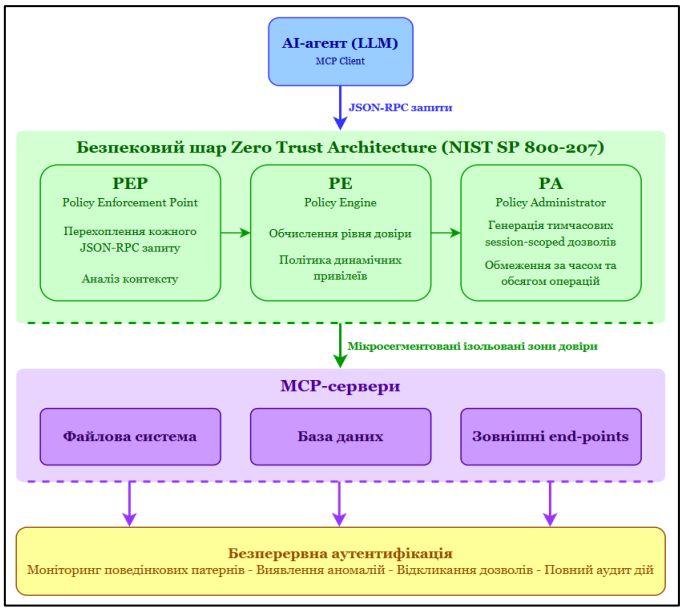


Рисунок 1 – Архітектурна схема інтеграції Zero Trust Architecture у MCP-агентні системи

Запропонований підхід інтегрує перевірені принципи Zero-Trust з урахуванням специфіки MCP-протоколу. На відміну від існуючих рішень, що зосереджуються на мережевому рівні безпеки, запропонований метод оперує на рівні семантики дій AI-агента, що забезпечує адекватний захист від загроз, специфічних для LLM-агентних систем. Подальша робота передбачає програмну реалізацію прототипу та експериментальну оцінку впливу безпекового шару на латентність та повноту функціональності агента.

### *Список використаних джерел:*

1. Soria Parra D., Spahr-Summers J. Model Context Protocol Specification (version 2025-11-25). Anthropic, 2024–2025. URL: https://modelcontextprotocol.io/specification/2025-11-25 (дата звернення: 01.03.2026).

2. Hasan M. M., Li H., Fallahzadeh E. та ін. Model Context Protocol (MCP) at First Glance: Studying the Security and Maintainability of MCP Servers. arXiv. 2025. arXiv:2506.13538.

3. Hou X., Zhao Y., Wang S., Wang H. Model Context Protocol (MCP): Landscape, Security Threats, and Future Research Directions. arXiv. 2025. arXiv:2503.23278.

4. He Y., Wang E., Rong Y. та ін. Security of AI Agents. arXiv. 2024. arXiv:2406.08689.

5. Rose S., Borchert O., Mitchell S., Connelly S. Zero Trust Architecture. NIST Special Publication 800-207. Gaithersburg : National Institute of Standards and Technology, 2020. 59 p. DOI: 10.6028/NIST.SP.800-207.