Формалізація атак підміни інструкцій у великих мовних моделях та методи їх виявлення

УДК 004.8:004.056.5

Роман Шклярський[1], Даниїл Журавчак[2]

Національний університет "Львівська політехніка",
[1]roman.shkliarskyi.asp.2025@lpnu.ua, [2]danyil.y.zhuravchak@lpnu.ua

Атаки типу prompt injection потрапили до переліку OWASP Top 10 для LLM-застосунків у 2025 році як основна загроза [1]. Причина полягає в архітектурній особливості трансформера: модель не розрізняє керуючі інструкції оператора та дані користувача, що дозволяє зловмиснику підмінити поведінку системи через текстовий вхід. Традиційні метрики оцінювання вразливостей, зокрема CVSS, не враховують стохастичну природу LLM, тому виникає потреба у формалізованих підходах до класифікації та вимірювання таких атак [2].

Перший систематичний підхід до формалізації запропонували Liu et al. у дослідженні, представленому на USENIX Security 2024 [3]. Атака описується як задача оптимізації: зловмисник прагне максимізувати ймовірність цільової відповіді R при заданому запиті Q та впровадженій інструкції. Для вимірювання ефективності атак введено метрику Attack Success Variation (ASV), яка дозволяє кількісно порівнювати п'ять типів атак із десяти різних механізмів захисту на десяти провідних моделях.

Паралельно з формалізацією дослідники розробляли таксономії для класифікації атак. Робота SoK, опублікована у 2026 році, вводить тривимірну таксономію за векторами доставки (прямі та непрямі), модальностями атаки (текстові та обфусковані) і поведінкою поширення (persistent та transient) [4]. Аналіз понад 30 CVE у комерційних агентах, зокрема Claude Code та GitHub Copilot, показав, що непрямі атаки через протокол MCP призводять до виконання довільного коду та витоку облікових даних. Окремо виділено клас атак "Confused Deputy", де агент використовує власний авторизований доступ до інструментів для виконання дій на користь зловмисника [4].

Дослідження OpenClaw розширює таксономію до трьох рівнів: ланцюжок постачання, активація та виконання [5]. Серед семи категорій загроз найкритичнішою є вихід із ізольованого середовища, для якого базовий рівень захисту становить лише 17% [5]. Атаки через кодування (Base64, шістнадцяткове представлення) обходять статичні фільтри і класифіковані як середня загроза за MITRE ATLAS AML.TA0008 [5].

Серед архітектурних методів захисту виділяються два підходи з підтвердженою ефективністю. Система StruQ розділяє вхідні дані на привілейований канал інструкцій і ненадійний канал даних за допомогою зарезервованих роздільників та спеціалізованого налаштування моделі [6]. Підхід SecAlign формулює захист як задачу оптимізації переваг через Direct Preference Optimization: модель навчається на трійках {вхід, бажана відповідь, небажана відповідь}, що дозволяє розрізняти оригінальні інструкції від зовнішніх даних [6].

Таблиця 1

Порівняння методів захисту від атак підміни інструкцій

Метод захисту	Технічний підхід	Середній рівень захисту (ASV)	Основне обмеження
StruQ	Розмежування каналів за роздільниками	95.0% -- 98.0%	Потребує переналаштування моделі
SecAlign	Оптимізація переваг (DPO)	~100.0%	Вразливий до адаптивних алгоритмів
InstruCoT	Міркування за ланцюжком думок (CoT)	92.5%	Збільшена затримка виведення
HITL	Перехоплення людиною в контурі	19% → 92.0%	Вузьке місце для автономних задач

Для відомих векторів атак ASR наближається до 0%, хоча метод залишається вразливим до адаптивних алгоритмів. Порівняння методів захисту наведено у табл. 1. Метрика ASV, запропонована Liu et al., дозволяє стандартизувати порівняння захисних рішень, проте не враховує ефект підсилення ризику через автономність агента [3]. Наведені дослідження показують, що формалізація атак типу prompt injection перейшла від окремих описів до кількісних фреймворків з уніфікованими метриками. Таксономії SoK та OpenClaw охоплюють як текстові, так і агентні вектори атак, а AIVSS забезпечує інструмент для оцінювання ризику в контексті конкретного розгортання. Архітектурні підходи StruQ і SecAlign демонструють вищу ефективність, проте обидва вимагають переналаштування моделі, що обмежує їх застосування з комерційними закритими системами. Стандартизація бенчмарків оцінювання залишається відкритою проблемою.

1. LLM01:2025 Prompt Injection. OWASP Gen AI Security Project. 2025. URL: https://genai.owasp.org/llmrisk/llm01-prompt-injection/ (дата звернення: 05.05.2026).
2. AIVSS Scoring System For OWASP Agentic AI Core Security Risks v0.8. OWASP Agentic AI Security Working Group. 2025.
3. Liu Y. et al. Formalizing and Benchmarking Prompt Injection Attacks and Defenses. USENIX Security Symposium. 2024.
4. Prompt Injection Attacks on Agentic Coding Assistants: A Systematic Analysis of Vulnerabilities in Skills, Tools, and Protocol Ecosystems. arXiv. 2026.
5. Towards Secure Agent Skills: Architecture, Threat Taxonomy, and Security Analysis. IEEE S&P. 2026.
6. Chen S. et al. StruQ: Defending Against Prompt Injection with Structured Queries. USENIX Security Symposium. 2025.