УДК 004.056

From Vulnerability to Vector: How Prompt Injection Enables Broader Cyber Threats in LLM-Integrated Systems

Ali Mahmoud Alkouny (masters student),
Scientific supervisor –Savkov S.V.
ITMO University
alimah2001@hotmail.com

Abstract
Prompt injection has been characterized by OWASP as one of the most critical vulnerabilities in LLM-based applications [4]. Traditionally, it has been considered in the context of jailbreaking or generating offensive content. However, with the evolution of LLMs from simple chatbots to more intelligent agents that can execute API calls, interact with databases, or even execute shell commands [2, 5], the threat model associated with this type of vulnerability is entirely different. This shift means that prompt injection can now lead to unauthorized actions such as data exfiltration, system compromise, or lateral movement within networks, making it a more severe security threat [3]. This paper presents a new model, which we term the "Injection-to-Compromise" (ItC) model, for characterizing the potential of prompt injection attacks as a type of initial vector in a complex cyberattack model [4]. This model categorizes injection techniques based on their potential for post-exploitation activities and maps them to the privileges of the integrated systems [4]. By synthesizing the research associated with the OWASP Top 10 vulnerabilities for LLMs, IBM's research on the risks associated with prompt injection attacks, and the systematic research on LLMs in the context of cybersecurity applications [1], we have developed a new model that characterizes the potential for post-exploitation activities associated with different injection techniques [4]. This paper has found that there is a clear distinction between direct injections that can execute arbitrary commands with authorization privileges and indirect injections that can be embedded in different sources such as emails or webpage content for the purpose of executing a type of "worm"-type attack [4]. This distinction is crucial because direct injections typically require user interaction, whereas indirect injections can automatically trigger malicious actions when the LLM processes contaminated data, potentially spreading like a worm across connected systems [4]. This paper has found that the potential for a prompt injection attack is directly related to the privileges associated with the controlling application rather than the LLM model [1]. A defense-in-depth model is presented that characterizes the potential for mitigating this type of vulnerability through the use of the least privilege model, API control, and separation of instructions and data [4].

Keywords
Prompt Injection, LLM Security, Cyber Kill Chain, Indirect Prompt Injection, LLM Integration Data Exfiltration, Remote Code Execution, Privilege Escalation

Literature

1. Touvron H., Lavril T., Izacard G., Martinet X., Lachaux M.-A., Lacroix T., Rozière B., Goyal N., Hambro E., Azhar F., et al. Llama: Open and efficient foundation language models // arXiv preprint arXiv:2302.13971. – 2023.

2. Ge Y., Hua W., Mei K., Tan J., Xu S., Li Z., Zhang Y., et al. Openagi: When llm meets domain experts // Advances in Neural Information Processing Systems. – 2024. – Vol. 36.

3. Ghelani D. Cyber security, cyber threats, implications and future perspectives: A review // Authorea Preprints. – 2022.

4. Pankajakshan R., Biswal S., Govindarajulu Y., Gressel G. Mapping llm security landscapes: A comprehensive stakeholder risk assessment proposal // arXiv preprint arXiv:2403.13309. – 2024.

5. Gemini Team, Anil R., Borgeaud S., Wu Y., Alayrac J.-B., Yu J., Soricut R., et al. Gemini: A family of highly capable multimodal models // arXiv preprint arXiv:2312.11805. – 2023.

Author: Ali Mahmoud Alkouny

Scientific supervisor: Savkov S.V.