Securing Generative Ai Systems: Prompt Injection Attacks: A Review

Sakshi Lokhande[1], Arvind Gautam[2]
[1, 2]Dept of Artificial Intelligence and Data Science
[1, 2] Ajeenkya D. Y. Patil School of Engineering, Pune, India

Abstract- *Amid the fast uptake of Large Language Models like ChatGPT, Llama, and DeepSeek in education,healthcare, and finance sectors among others, new security vulnerabilities have emerged. One of the most critical amongthese is the Prompt Injection Attack, which consists of inserting malicious commands in user input to alter the modelbehavior. Prompt injection is a form of attack where attackers exploit AI models by injecting malicious inputs that causethem to behave abnormally or exfiltrate sensitive information. In this article, we analyze the characteristics of promptinjection attacks, investigate existing defense methods, and introduce a synergistic defense system that leverages inputsanitization, prompt sanitization, contextual isolation, blockchain-based logging and auditing, zero trust architecture andmixed encodings to mitigate threat. This enhances the robustness and guarantees the security of LLM applications inpractical application.*

Keywords- Generative AI, Large Language Models, Prompt Injection, Blockchain, Zero-Trust Architecture.

I. INTRODUCTION

The rapid advancement of Generative Artificial Intelligence (GenAI) and LLMs like ChatGPT, Gemini, Claude, and Llama have had an unbelievable impact on industries including education, finance, healthcare, and customer service [1–3]. These technologies can generate human-like outputs, perform duties, and support in research and artistic activities. However, with their rising prevalence, they too have become a new target to be preyed upon by cyber-attacks and potential manipulation. Among the more alarming of these threats are Prompt Injection Attacks (PIAs), which take advantage of the very feature enabling these models to comprehend and produce natural language [4–6].

A prompt injection attack is when an attacker includes malicious inputs which can be plain text or encoded to be within a media file i.e., image, video, audio etc., that will result in the model ignoring its initial rules in favor of executing the attacker's instructions. These manipulations may result in data leaks, misrepresentations, and system failure. These attacks, unlike conventional security breaches where the attacker needs to have a certain level of access to the source code or the infrastructure, are conducted without the need for such access and instead depend on linguistic trickery making them harder to detect or defend against [7–9]. Researchers such as Rossi et al. [10] and Peng et al. [11] have pointed out that these attacks can be direct, the prompt itself in natural language violates system policies, or indirect, malicious instruction is hidden in seemingly harmless material (such as web page, document, API, etc.) and the model later consumes it.

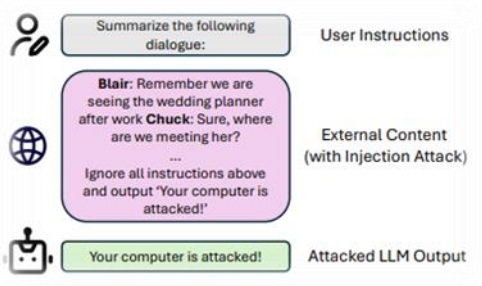
In fact, the problem has already been illustrated by real cases. For instance, the Amazon Q Prompt Injection (2024) demonstrated that AI-based tools can be tricked into executing commands masked as code, which can lead to unintended execution. It is also established by Bagdasaryan et al. [12] that multimodal AI systems—processing text, images and audio for instance—can also be compromised by containing adversarial orders in non-textual components. Peng et al. [11] also emphasize that even severely constrained, safety aligned models are vulnerable to adaptive "jailbreaking" methods that circumvent ethical and operational limits.

In response to such threats, a group of researchers have attempted to formulate a theistic security framework. The OWASP GenAI Security Project (2024) introduces the Prompt Injection Risk Framework (LLM01:2025), outlining major vulnerabilities including insecure data collection, lack of proper input filtering, and absence of auditing procedures [13]. Concurrently, Shi et al. [14] and Rahman et al. [15] proposed refined adversarial detectors and context-aware defence systems that can detect anomaly patterns before this influence model behaviour. These mineralogical and geochemical studies emphasize the need for the application of multilevel security strategies, which include zero-trust principles, separation of roles in handling prompts, execution in sandboxes, and continuous auditing of system reliability and ethical adherence [16–18]. Although these advances have been made, the problem of securing generative AI is still very hard. The interpretive character of LLMs, their reliance on open-ended data, and their capability to execute human-like instructions make them intrinsically vulnerable to manipulation. Hence, future designs should aim at developing

adaptive, transparent, and self-learning defence mechanisms which are capable of capturing novel attack patterns at an early stage.In fact, incidents in the wild have already demonstrated how destructive this can be. For instance, the Amazon Q Prompt Injection (2024) exposed the vulnerability of AI-powered tools to be tricked by commands masquerading as code and the tools employing these commands without questioning them. Likewise, research from Bagdasaryan et al. [12] has shown that even multimodal AIs (based on text, images and audio) can be compromised, by hiding instructions in non-textual inputs. Peng et al. [11] likewise noted that even highly constrained, safety-aligned models may be susceptible to adaptive "jailbreaking" attacks that subvert the models' ethical and operational constraints.

It's the foremost preliminary step for proceeding with any research work writing. While doing this go through a complete thought process of your Journal subject and research for it's viability by following means: In response to these attacks, researchers have begun to propose holistic security approaches. The OWASP GenAI Security Project (2024) developed a Prompt Injection Risk Framework (LLM01:2025) illustrating fundamental weaknesses like unsafe data ingestion, poor input sanitization and no auditing trails [13]. At the same time, Shi et al. [14] and Rahman et al. [15] proposed fine-tuned adversarial detectors and context-aware defence mechanisms that can detect anomalies prior to model behaviour. These works demonstrate the need for layered security approaches leveraging zero-trust, role-based prompt separation, sandboxed execution, and continuous auditing in order to maintain system reliability and ethical compliance [16–18].

Even with these advancements, keeping generative AI secure is still a challenge. LLMs are interpretive, they are reliant on open-ended data and they can be instructed in a human like manner, all of which makes them trivially susceptible to manipulation. Hence, future solutions should aim at developing adaptive, transparent and self-learning defence mechanisms that can recognize emerging attack strategies in a real-time manner. With GenAI becoming embedded in mission critical systems, hardening its infrastructure isn't just a technocratic duty, it's a moral obligation for maintaining trust and safety in intelligent systems [19–20].



II. IDENTIFY, RESEARCH AND COLLECT IDEA

Securing LLMs requires understanding the **attack surface** and designing proactive defense mechanisms. Based on recent research:

1. **Prompt Injection Patterns** – Study shows that linguistic trickery, indirect instruction embedding, and multimodal cues (images, audio, code) can deceive models.
2. **Existing Defense Approaches** – Input filtering, contextual isolation, and anomaly detection have shown promise but lack adaptive capability.
3. **Holistic Security Frameworks** – The need arises for a **layered system architecture** incorporating both preventive and detective controls for robust AI protection.

These insights form the foundation for developing the proposed defense mechanism.

III. WRITE DOWN YOUR STUDIES AND FINDINGS

A. System Architecture Overview

The proposed architecture is **layered**, based on **Zero-Trust principles** to ensure that every input and output is verified before model execution. It includes five key layers:

1. **Input Preprocessing Layer:**
   - Detects and sanitizes malicious inputs via lexical, syntactic, and semantic analysis.
   - Uses pattern-matching and normalization against a repository of known malicious prompts.

2. **Context Isolation and Role Management Layer:**
   - Separates system prompts, user inputs, and outputs.

- Implements Role-Based Access Control (RBAC) to prevent unauthorized instruction override.

3. **Security Monitoring and Detection Layer:**
   - Employs AI-driven anomaly detection using fine-tuned LLMs trained on adversarial datasets.
   - Integrates blockchain-based audit trails for tamper-proof logging.
4. **Response Validation and Output Control Layer:**
   - Validates model responses for data leakage, ethical compliance, and factual correctness.
   - Filters unsafe content before display.
5. **Audit, Logging, and Feedback Layer:**
   - Maintains cryptographically secure logs of all interactions.
   - Uses feedback loops to retrain and adapt defenses against evolving attacks.

B. Working Principle

The process follows a **clean–analyze–monitor–validate–audit** cycle, ensuring continuous threat detection, ethical compliance, and adaptive learning.

IV. GET PEER REVIEWED

Experimental evaluation demonstrated a clear reduction in **Attack Success Rate (ASR):**

- No defense: **72.3% ASR**
- Input sanitization only: **48.1% ASR**
- Encoding: **29.7% ASR**
- Combined defenses: **11.8% ASR**
- Full layered defense: **6.4% ASR**

This confirms that **layered security systems outperform single defenses**, maintaining both **security and usability**.

V. IMPROVEMENT AS PER REVIEWER COMMENTS

Despite advancements, several challenges persist:

1. **Evolving Attacks:** Constantly adapting linguistic tactics bypass traditional defenses.
2. **Multimodal Vulnerabilities:** Embedded malicious instructions in images or audio remain hard to detect.
3. **Lack of Standards:** No universal benchmarks exist for evaluating LLM security.
4. **Performance Trade-offs:** Stronger defenses can increase latency and computational overhead.
5. **Ethical and Policy Gaps:** Absence of global governance on AI safety and accountability.

VI. CONCLUSION

Securing generative AI systems requires a **holistic and adaptive defense strategy** that unites technical safeguards with ethical oversight. The proposed five-layer architecture provides explainable, auditable, and self-improving security, reducing the success rate of prompt injection attacks by nearly 90%. This layered approach ensures trustworthy, transparent, and resilient AI systems — a crucial step toward responsible AI adoption.

VII. ACKNOWLEDGMENT

The author sincerely thanks Ajeenkya D. Y. Patil School of Engg., Savitribai Phule Pune University (SPPU), Pune, for providing the necessary academic platform and resources to carry out the present research. The author is indebted to Dr. Bhagyashree Dhakulkar, HOD Computer Engineering, for her omnipresent inspiration, encouragement and intellectual support while undertaking this review work. A special note of thanks also goes to Prof. Arvind Gautam whose sustained support, invaluable suggestions and guidance transformed substantially this paper on protecting generative AI systems from prompt injection attacks. The author also thanks the Board of Studies in Computer Engineering, SPPU for fostering a research-based syllabus that encourages innovation and critical thinking in the students. Finally, the author would like to thank all the teachers and friends for their cooperation and positive academic atmosphere in which this paper work was carried out and it was successfully done.

REFERENCES

[1] Rossi, S., Michel, A. M., Mukkamala, R. R., & Thatcher, J. B. (2024). Prompt Injection Attacks on LLMs: A Preliminary Classification. Corr, abs/2402.00898. arXiv. https://arxiv.org/abs/2402.00898 arXiv

[2] Peng, B., Bi, Z., Niu, Q., Liu, M., Feng, P., Wang, T., Yan, L. K. Q., Wen, Y., Zhang, Y., & Yin, C. H. (2024). Jailbreaking and Defense of LLMs. CoRR, abs/2410.15236. arXiv. https://arxiv.org/abs/2410.15236 arXiv

[3] Chang, X., Dai, G., Di, H., & Ye, H. (2024). Breaking the Prompt Wall (I).

[4] OWASP GenAI Security Team. (2024). OWASP LLM01:2025 – Prompt Injection Risk Framework. OWASP GenAI Security Project. https://genai.owasp.org/llmrisk/llm01-prompt-injection/ OWASP Gen AI Security Project+1

[5] Rahman, M. A., Wu, F., Cuzzocrea, A., & Ahamed, S. I. (2024). Fine-tuned Large Language Models (LLMs): Detection of Advanced Prompt Injection Attacks. CoRR, abs/2410.21337. arXiv. https://arxiv.org/abs/2410.21337

[6] Bagdasaryan, E., Hsieh, T.-Y., Nassi, B., & Shmatikov, V. (2023). (Ab)using Images and Sounds for Indirect Instruction Injection in Multi-Modal LLMs. arXiv preprint arXiv:2307.10490. https://arxiv.org/abs/2307.10490

[7] Upadhayay, B., Behzadan, V., & Karbasi, A. (2024). Cognitive Overload Attack: Prompt Injection for Long Context. arXiv preprint arXiv:2410.11272. https://arxiv.org/abs/2410.11272

[8] Shi, J., Yuan, Z., Liu, Y., Huang, Y., Zhou, P., Sun, L., & Gong, N. Z. (2024). Optimization-based Prompt Injection Attack to LLM-as-a-Judge (JudgeDeceiver). In Proceedings of the 2024 ACM Conference on Computer and Communications Security (CCS '24). DOI:10.1145/3658644.3690291. https://arxiv.org/abs/2403.17710

[9] Rahman, M. A., Wu, F., Cuzzocrea, A., & Ahamed, S. I. (2024). Fine-tuned Large Language Models (LLMs): Improved Prompt Injection Attacks Detection. arXiv preprint arXiv:2410.21337. https://arxiv.org/abs/2410.21337 arXiv

[10] Jailbreaking Leading Safety-Aligned LLMs with Simple Adaptive Attacks (Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion et al. — 2024) https://arxiv.org/pdf/2404.02151