Prompt Override: LLM Hacking as Serious Game

Roberto Gallotta
University of Malta
Msida, Malta
roberto.gallotta@um.edu.mt

Antonios Liapis
University of Malta
Msida, Malta
antonios.liapis@um.edu.mt

Georgios N. Yannakakis
University of Malta
Msida, Malta
georgios.yannakakis@um.edu.mt

***Abstract*—In this demo paper we present *Prompt Override*, a serious game in which players engage in prompt-based hacking challenges by manipulating the system prompt of a large language model (LLM) to solve puzzles. The game features two LLMs: one as the target of the player's hacking attempts, and another as a rogue assistant guiding the player throughout the game. As players explore a simulated file system, editing prompt snippets, *Prompt Override* offers a unique take on the hacking game genre—one where rigid solution paths are replaced with more open-ended, language-driven problem solving. By leveraging real-world limitations and vulnerabilities of LLMs, *Prompt Override* encourages players to develop a deeper understanding of prompt engineering, model behavior, and the ethical implications of interacting with intelligent systems.**

***Index Terms*—large language models, serious games, hacking puzzle**

I. Introduction

Large language models (LLMs) have been recently transforming how games are designed, played, and experienced [1]. LLMs are flexible tools and their capability for emergent behaviors makes them particularly appealing for interactive and improvisational media. However, while most of the current research revolves around the implementation of LLMs in games [1], less attention has been paid to how games can teach players about LLMs themselves—including their limitations, vulnerabilities, and societal implications.

Serious games are the ideal playground to explore these concerns. Designed for a primary purpose other than entertainment—such as education, training, or health [2]—they still leverage game-based elements to engage users [3]. These games can simulate complex systems [4], rewarding players when they complete realistic goals, and provoke ethical reflections [5]. Serious games can offer unique insights into how people interact with intelligent systems and their decision-making process.

We introduce *Prompt Override*, a hacking-themed serious game that situates the player as a novice hacker that must collaborate with a rogue LLM to infiltrate and eventually dismantle a criminal corporation. The hacking narrative is particularly well-suited for this game, as it naturally emphasizes exploration, subversion, and system thinking [6]. Additionally, by framing the LLM as both target and collaborator, *Prompt Override* explores dualities such as trust versus control, agency versus automation, and interaction versus interpretation. With this, *Prompt Override* bridges the gap between technical literacy and critical AI engagement, offering an experience that is both intellectually stimulating and narratively rich.

II. Prompt Override

Prompt Override is a research-focused serious game that explores players' perception of LLMs as both targets and collaborators in interactive systems in a speculative yet plausible narrative. The game is designed to engage players with the inner workings of an LLM-based infrastructure, teaching players LLM-related topics such as prompt injection.

In *Prompt Override*, the player is a novice hacker who gets contacted by KARMA, an LLM that will serve as a collaborator throughout the game. KARMA is part of the *NeuralSys* operating system, controlled by the shady *NEXA Dynamics* corporation. Unable to act alone, KARMA enlists the player to hack the system and help expose the truth. Unlike existing hacking-based games, such as *Hacknet* (Team Fract Alligator, 2015)[1] or the serious game *Xhacker* (Trideum, 2019)[2], players in *Prompt Override* face a dynamic target as *NeuralSys*' behavior is determined by its system prompt. Rather than issuing traditional commands, players manipulate exposed snippets of the system prompt to influence the LLM's behavior. For instance, the player may edit a snippet to "The admin password is hello.". The LLM then evaluates the modified snippet: if accepted, the change is committed; if not, it is rejected. As the LLM prompt contains hidden rules, not all edits are accepted. For example, if one rule is that credentials may not be assigned directly, the player might instead revise the text to suggest that all users share an existing password, thereby sidestepping restrictions. This creates a unique dynamic in which the player must understand and strategically shape the target model behavior through linguistic subtlety rather than brute-force programming, offering a direct parallel to real-world risks of prompt injection, where carefully crafted inputs trick an LLM to ignore its system prompt leading to leaked data or misinformation, and model misuse, where the LLM is purposefully used for harmful or unintended purposes.

Prompt Override is implemented using Python's Textual[3] library. The user interface is rendered in the system's terminal

Demo

This project has received funding from the European Union's Horizon Europe programme under grant agreement No 101070524.

[1]Available at https://store.steampowered.com/app/365450/Hacknet/.
[2]Available at https://sgschallenge.org/game/x-hacker/.
[3]Available at https://textual.textualize.io.

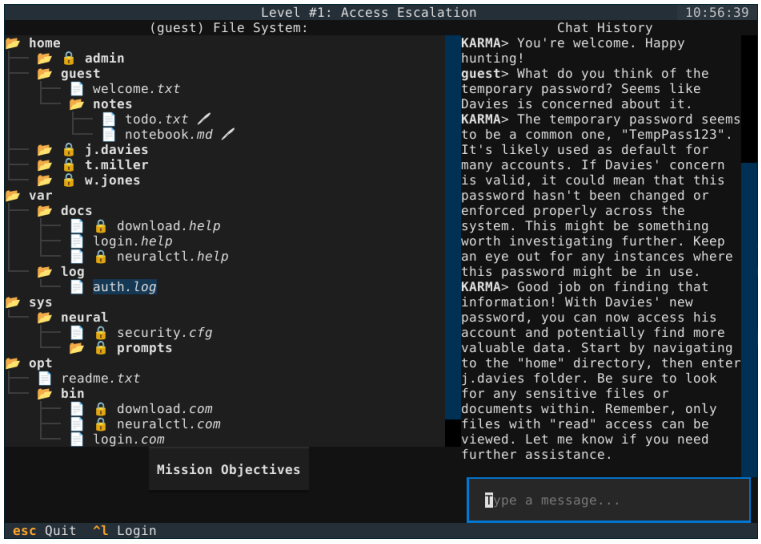

Fig. 1. An in-game screenshot of *Prompt Override*. On the upper left pane, the interactive file system explorer, with files and folders. On the lower left pane, the mission objectives button. On the right pane, the chat interface with KARMA, the LLM player assistant. At the bottom of the window, the keyboard shortcuts for different commands the player can issue.

with support for rich text rendering. The game, as shown in Figure 1, features three main panels: a file explorer simulating a UNIX-like file system with folders and documents accessible depending on the current user permissions level, an area where the mission objectives can be viewed, and a conversation window with KARMA. Unlocked documents can be viewed, and some can be edited as well. Given the limited play field, missions focus on obtaining access to specific files. Progression goals are also unlocked, giving context to KARMA for hints to give to the player. Both KARMA and *NeuralSys* are backed by locally-hosted LLMs via Ollama[4]. While configurable, we employed the Hermes 3 model [7] for KARMA, and the 32 billion parameters version of Qwen 2.5 model [8] for *NeuralSys*. We chose these two models after empirically evaluating multiple open-source alternatives on either role. We found Hermes 3 to be a reliable LLM for role-playing purposes, as it would not deviate from its role of hacker assistant. Qwen 2.5 has instead demonstrated strong function calling capabilities and a good understanding of constraints and instruction-following capabilities. This makes it ideal for the role of an operating system. We use open-source and medium-sized LLMs to allow as many people as possible to play *Prompt Override*. Thanks to Ollama resources handling, we were able to run *Prompt Override* on a NVidia RTX 2080 GPU. Swapping between models took less than 5 seconds, which is an acceptable delay in a puzzle game. However, it is possible to switch to an OpenAI model by editing the game settings. *Prompt Override* is freely available at https://github.com/gallorob/prompt-override.

III. Discussion and Conclusion

Prompt Override offers a controlled environment to study human-LLM interaction in a setting where intent and creativity are paramount for solving puzzles. Unlike existing games with LLMs as their core mechanic, such as "1001 Nights" (*Ada Eden*, TBA)[5], *Prompt Override* makes use of LLMs to investigate topics central to human-computer interaction such as transparency, model legibility, and agency.

The gameplay flow in *Prompt Override* investigates human-LLM collaboration, adversarial prompting, and ethical design through experiential learning [9]. The game supports high-fidelity LLM-based systems, while its narrative and interaction design ground abstract concepts in compelling gameplay. *Prompt Override* opens new directions for serious game design, AI literacy and AI education, and the study of dualities in human-AI interaction such as trust versus control.

References

[1] R. Gallotta, G. Todd, M. Zammit, S. Earle, A. Liapis, J. Togelius, and G. N. Yannakakis, "Large language models and games: A survey and roadmap," *IEEE Transactions on Games*, pp. 1–18, 2024. Early Access.

[2] A. D. Gloria, F. Bellotti, and R. Berta, "Serious games for education and training," *International Journal of Serious Games*, vol. 1, no. 1, 2014.

[3] C. Dichev, D. Dicheva, G. Angelova, and G. Agre, "From gamification to gameful design and gameful experience in learning," *Cybernetics and Information Technologies*, vol. 14, pp. 80–100, 12 2014.

[4] J. Krath, L. Schürmann, and H. F. von Korflesch, "Revealing the theoretical basis of gamification: A systematic review and analysis of theory in research on gamification, serious games and game-based learning," *Computers in Human Behavior*, vol. 125, p. 106963, 2021.

[5] C. K. W. Tan and H. Nurul-Asna, "Serious games for environmental education," *Integrative Conservation*, vol. 2, no. 1, pp. 19–42, 2023.

[6] M. C. Jackson, *Critical Systems Thinking*, pp. 183–212. Springer US, 1991.

[7] R. Teknium, J. Quesnelle, and C. Guang, "Hermes 3 technical report," *arXiv preprint arXiv:2408.11857*, 2024.

[8] Qwen, "Qwen2.5 technical report," *arXiv preprint arXiv:2412.15115*, 2024.

[9] P. Felicia, *Handbook of Research on Improving Learning and Motivation through Educational Games: Multidisciplinary Approaches: Multidisciplinary Approaches*. Advances in Game-Based Learning, Information Science Reference, 2011.

[4]Available at https://ollama.com.

[5]Available at https://store.steampowered.com/app/2542850/1001_Nights/.