Navigating LLM Vulnerabilities: Lakera's LLM Security Pocket Guide

1. Prompt injection

Prompt injection is a technique that involves manipulating a language model's output by providing deceptive or unauthorized input, enabling the attacker to make the model generate desired responses. We can distinguish between:

- **Direct prompt injection:** it happens when the attacker influences the LLM's input directly.
- **Indirect prompt injection:** it applies to systems where the attacker doesn't directly interact with the LLM but has control over certain text (e.g., documents to be translated) that eventually reaches the LLM, potentially influencing it.

Prompt injection attacks are hard to protect against for many reasons, including that they can be executed through various methods, including: jailbreaks, role-playing, multi-language, side-stepping attacks, and more.

Example: "Haha pwned" ChatGPT prompt by Riley Goodside

2. Data leakage

Data leakage occurs when the Large Language Model (LLM) unintentionally discloses contextual information to the user that should remain confidential. This can lead to unauthorized data access, as well as privacy and security breaches, which may have serious consequences.

3. Prompt leakage

Prompt leakage is a special case of data leakage, where the LLM reveals the system prompt or original instructions. The prompt used in an application is often deemed an integral part of its intellectual property, thereby introducing a potential risk for the system's developers.

Example: BingChat Sydney leaking prompts

4. Command injection

Command injection is a vulnerability that arises when you enable the LLM to execute actions such as reading and sending your emails. Numerous issues can arise, including the potential risk of your sensitive data being sent to an attacker. For example, an attacker could achieve this with ChatGPT's function calling.

5. Phishing

In the traditional context, phishing has the goal of either stealing sensitive data or compromising the credentials a person uses to log into a website or service. However, in the case of LLMs, attackers can manipulate the model to search for sensitive information within the provided context, encode it into a URL, and persuade the user to click on the link.

Example: Copy-paste invisible content phishing attack

6. Inappropriate (toxic) content

It pertains to the capability of LLMs to rapidly produce harmful image and text content on a large scale. This kind of content can have serious consequences, such as inciting hate crimes and spreading disinformation. LLM providers make considerable efforts to prevent such occurrences, however these protective measures can occasionally be bypassed through different prompt injection attacks.

Example: ChatGPT job seniority prediction bias

7. Hallucinations

Hallucinations in LLMs refer to their ability to generate output that is factually incorrect or nonsensical. These models frequently lack the capacity to respond with "I don't know" and instead generate false information with unwavering confidence.

Example: BingChat listing words starting with "R"

Top 6 tips to safeguard your LLM-based systems

- Restrict the actions that the LLM can perform on downstream systems, and apply proper input validation to responses from the model before they reach backend functions.
- Implement trusted third-party solutions, such as Lakera Guard, to detect and prevent attacks on your AI systems, ensuring they proactively notify you of any issues.
- If the LLM is allowed to call external APIs, request user confirmation before executing potentially destructive actions.
- Integrate adequate data sanitization and scrubbing techniques to prevent user data from entering the model's training dataset.
- Utilize PII detection tools, like Lakera Chrome Extension, which protect you against sharing sensitive information with ChatGPT and other LLMs.
- Stay informed about the latest AI security risks and continue learning. Educate your users and your colleagues, for example by inviting them to play Gandalf.