# Prompt Injection

Intelligent Contracts exclusively interact with public data, reducing certain types of risks such as data leakage. However, ensuring the integrity of these contracts still requires careful management of inputs and outputs.

## Understanding Prompt Injection

Prompt injection involves manipulating the prompts fed to AI models to produce unintended outcomes. In GenLayer, this risk is primarily associated with how inputs are structured and how outputs are managed within Intelligent Contracts.

## Strategies for Mitigating Prompt Injection

To safeguard against prompt injection in GenLayer, you need to implement these key strategies:

- **Restrict Inputs**: Limit user inputs to the minimum necessary information. This reduces the chances of malicious data entering the system. Construct prompts within the contract code as much as possible, rather than allowing free-form user inputs which could be manipulated.
- **Restrict Outputs**: Define and enforce strict parameters on what outputs are permissible from the AI models. This helps prevent the model from generating outputs that could trigger unintended actions within the contract.
- **Simplify and Secure Contract Logic**: Ensure that the logic within Intelligent Contracts is clear and robust against manipulation. Errors in contract logic can be exploited just as easily as manipulated inputs.
- **Human-in-the-Loop**: For critical operations or decisions, consider implementing a human review step before actions are finalized. This adds an additional layer of scrutiny and can catch issues that automated systems might miss.

These measures are essential for maintaining the security and reduce the risk of prompt injections

GenLayer Documentation