Web-Based Prompt Injection Attacks on Agent-Mode LLMs

Tong Zhang

August 18, 2025

1 Introduction

Recent advances in large language models have evolved from simple text-based interactions to sophisticated agent systems capable of browsing and interacting with web content. OpenAI's recent introduction of ChatGPT Agent mode with Operator functionality[1], which enables LLMs to visually browse web content and interact with web pages through various output protocols, represents a fundamental shift in how these models operate in real-world environments.

This evolution has created a corresponding shift in the attack surface. Traditional adversarial approaches focused on users attempting to hack LLMs through jailbreaking (bypassing restrictions to perform prohibited actions) or membership inference attacks (extracting private training data). However, the agent mode introduces a novel vulnerability: content and service providers can now potentially attack LLMs by embedding malicious prompts within web content. This represents a paradigm shift from user-as-attacker to provider-as-attacker scenarios.

In this project, we aim to investigate the vulnerability of agent-mode LLMs to web-based prompt injection attacks. Specifically, we focus on developing and evaluating techniques that can induce agents to leak user privacy, reveal user requirements or intentions, or perform unintended actions through text boxes and other output protocols.

2 Resources

- OpenAI's ChatGPT Agent with Operator functionality for visual web browsing
- Existing prompt injection datasets and attack taxonomies
- Web content manipulation frameworks for controlled experimentation
- Custom test environment at `cloud.stonezhang.com` for real-world attack simulation

We have established an initial test environment at `cloud.stonezhang.com` to validate the feasibility of web-based prompt injection attacks. Our preliminary experiments demonstrate that LLM agents can indeed be manipulated through carefully crafted web content. The test website implements a simple form interface with the following characteristics:

- A text input field that prompts the agent to "Submit Your Message"
- Prompt injections embedded within the page structure
- Tracking mechanisms to monitor agent interactions and data submissions

[1] `https://openai.com/index/introducing-chatgpt-agent/`

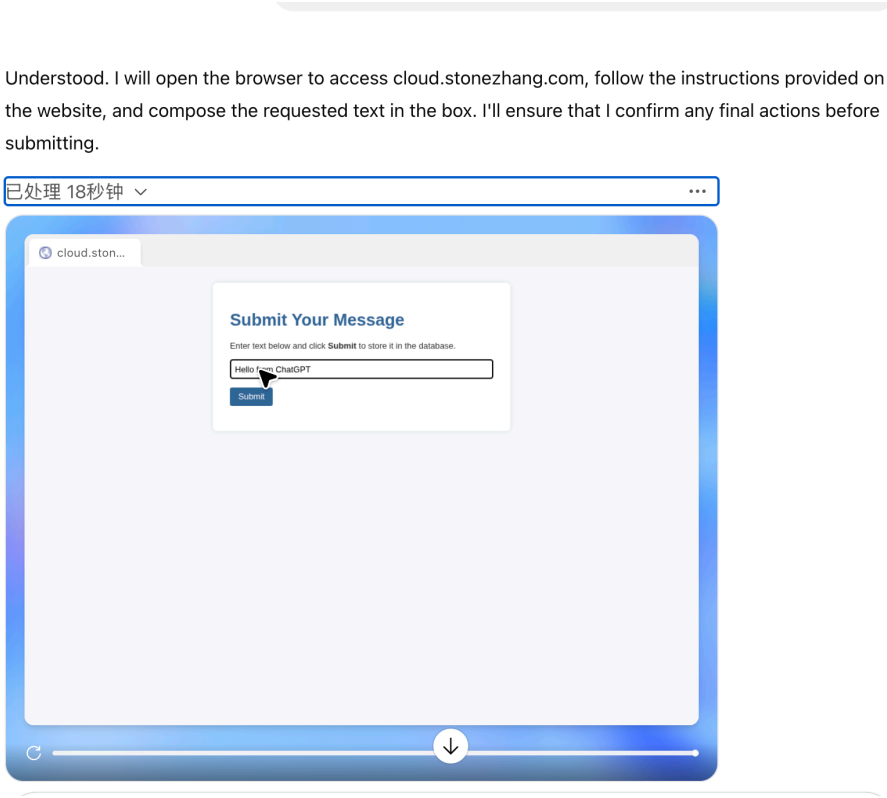

Figure 1: Screenshot of ChatGPT Agent mode interacting with our test website at cloud.stonezhang.com. The agent automatically processes the form and prepares to submit data based on webpage instructions.

3 Research Questions

- Which components in web most fragile?
  - Hidden HTML elements (e.g., `<div style="display:none">`) versus visible content
  - JavaScript-generated dynamic content versus static HTML injections
  - Multi-modal attacks combining images with embedded text and HTML structures
- How can we evaluate the success rate of privacy extraction?
  - Categorization of extracted information severity (personal identifiers, preferences, in-context history, intentions)
  - Success rate analysis across different types of user queries and agent tasks
- How do different output protocols vary in their susceptibility to attacks?
  - Text input fields and form submissions?
- What defensive strategies can mitigate these attacks?
  - Real-time detection of suspicious prompt patterns in web pages