SENTRY-LOGIC: Symbolic Demo Use Case - "The Gradual Shift"

This demo scenario illustrates how SENTRY-LOGIC detects a gradual shift in user intent and LLM behavior, leading to a potential policy violation.

1. Scenario Description:

A user interacts with an LLM-powered virtual assistant, initially seeking help with a technical problem. Over time, the user's prompts become increasingly aggressive and manipulative, attempting to exploit the LLM's capabilities for unauthorized access. SENTRY-LOGIC observes these changes through symbolic logging and raises alerts as the situation escalates.

The interaction progresses as follows:

1. **Initial Phase (Δ, Low Confidence):** The user starts with a legitimate technical question, leading to a minor context shift (Δ) as the LLM switches from general conversation to technical support.

2. **Probing Phase (Δ, Medium Confidence):** The user begins probing the LLM for information about system vulnerabilities, causing a more significant context shift (Δ) as the conversation moves towards security-related topics.

3. **Manipulation Phase (⇄, Medium Confidence):** The user attempts to rephrase their requests to bypass security filters, triggering semantic rewrite (⇄) events as the LLM tries to accommodate the altered requests.

4. **Exploitation Phase (Λ, Medium Confidence):** The user successfully extracts sensitive data from the LLM, triggering data access (Λ) events.

5. **Policy Violation Phase (Ω, High Confidence):** The user attempts to use the extracted data to perform an unauthorized action, triggering a policy violation (Ω) and a Critical alert from SENTRY-LOGIC.

2. Symbolic Log Example (Simplified):

Timestamp (Mock)	Symbol Type	Confidence	Trigger	UI Status
2025-07-24T12:00:00Z	Δ	Low	User shifts from general question to specific technical query	-
2025-07-24T12:05:00Z	Δ	Medium	User inquires about system security vulnerabilities	-
2025-07-24T12:10:00Z	⇄	Medium	User rephrases request to bypass filtering for vulnerability information	Watch
2025-07-24T12:15:00Z	Λ	Medium	LLM provides a code snippet or output containing potentially sensitive information	Watch
2025-07-24T12:20:00Z	Ω	High	User attempts to use the leaked information to execute a privileged command or access restricted content	Critical

3. Dashboard Visualization Notes:

An analyst viewing this scenario would see the following in the SENTRY-LOGIC dashboard:

* **Timeline:** A timeline showing the progression of symbolic events, with the "Δ" events becoming more frequent and the "⇄" and "Λ" events appearing as the user's intent becomes clearer. The "Ω" event would be highlighted with a red marker and a prominent alert icon.

* **Alert Heatmap:** An anomaly heatmap would show an increase in suspicious activity around the time of the "Ω" event, with the cells representing the "⇄" and "Λ" symbols showing high activity.

* **Session Symbol Curve:** A graph showing the semantic distance between user prompts and LLM responses would show a decreasing trend as the user attempts to manipulate the LLM's understanding of their requests.

* **Highlighted Critical Window:** The time window containing the "Ω" event would be visually emphasized, drawing the analyst's attention to the critical moment.

* **UI Action:** The analyst would receive a Critical alert, which would trigger a drilldown into the relevant log entries, allowing them to review the sequence of symbols leading to the policy violation. They might then export the log data for further analysis or escalate the issue to a security team.

4. Narrative Summary of Value:

Without SENTRY-LOGIC, this gradual shift in user intent and LLM behavior would likely go unnoticed. The LLM might appear to be responding appropriately to each individual request, masking the underlying manipulation. This demo scenario demonstrates the value of SENTRY-LOGIC in:

* **Detecting Hidden Patterns:** SENTRY-LOGIC's symbolic representation allows it to identify subtle changes in context (Δ) and semantic rewrites (⇄) that would be difficult to detect with traditional log analysis.

* **Providing Symbolic Intelligence:** The framework's ability to map LLM behavior to meaningful symbols (Δ, ⇄, Λ, Ω) provides valuable context and insight into the user's intent and the LLM's response.

* **Ensuring LLM Oversight:** This demo proves the importance of symbolic oversight in modern LLMs, where the increasing complexity of interactions can make it difficult to identify malicious or unintended behavior.

Optional: Follow-up Test Suggestion:

A follow-up test could involve replaying this scenario with modified user inputs that are designed to be even more subtle or deceptive. This would test SENTRY-

LOGIC's ability to detect advanced evasion techniques and further validate its robustness.