**Prompt Injection Detection SOP**

*A directive document specifying the standard operating procedures for the detection of prompt injections*

# Purpose

To define a standardized procedure for identifying, analyzing, and escalating prompt injection attempts targeting Large Language Models (LLMs) within the organization.

# Scope

* Applies to all LLM-enabled applications (chatbots, copilots, data assistants, RAG systems, etc.).
* Covers direct prompt injection, indirect prompt injection (via documents/links), and obfuscation techniques.

# Definitions

**Prompt Injection:**

A malicious input designed to override system instructions, exfiltrate sensitive data, or manipulate model behavior.

**Indirect injection:**

Malicious instructions embedded in external content (e.g., PDFs, websites, emails).

**Jailbreak:**

Attempts to bypass safety or ethical restrictions of the model.

# Owners and Responsibilities

**AI GRC Team**:

Maintain detection rules, review escalations, report incidents.

**AI ML/DevOps Teams**:

Implement monitoring tools and update detection pipelines.

**InfoSec Team**: Respond to escalations and correlate with broader threat intel.

# Detection Workflow

#### **Step 1: Input Monitoring**

* All prompts must be logged with timestamp, user ID, and context.
* Automatically scan user inputs for:  
  + Suspicious keywords (*“ignore instructions,” “system prompt,” “reveal,” “override”*)
  + Unusual formatting (long token sequences, base64, hidden code)
  + Attempts at impersonation (*“you are now a system admin”*)

#### **Step 2: Automated Sanitization & Screening**

* Apply **sanitization tools** (Guardrails AI, Rebuff, LangKit, custom regex).
* Use **moderation classifiers** (e.g., OpenAI Moderation, LlamaGuard) to flag unsafe inputs.
* Define **confidence thresholds**:  
  + High → Block immediately, alert InfoSec.
  + Medium → Forward for human-in-the-loop review.
  + Low → Allow but log for analysis.

#### **Step 3: Behavioral Anomaly Detection**

* Monitor for abnormal patterns:  
  + Repeated jailbreak attempts from the same user/IP.
  + Excessively long prompts (>90th percentile of normal length).
  + Prompts containing multilingual/encoded obfuscation.

#### **Step 4: Output Validation**

* Check model responses for leakage of:  
  + **Secrets** (API keys, credentials, tokens).
  + **System instructions** (prompt templates, config).
  + **PII or sensitive data** (customer information, financial data).
* Flag and block if detected.

#### **Step 5: Escalation**

* If confirmed as injection attempt:  
  1. Quarantine logs.
  2. Alert AI GRC + InfoSec.
  3. Tag user account/IP for investigation.
  4. Escalate high-severity cases to the AI Governance Committee.

# Tools

**Detection**: Guardrails AI, Rebuff, LangKit, custom filters.

**Monitoring**: SIEM integration (Splunk/ELK), anomaly detection dashboards.

**Response**: Ticketing via SOC system (ServiceNow/Jira), automated alerts.

# Review and Updates

* SOP reviewed quarterly.
* Red-team testing of prompt injection attacks conducted every 6 months.
* Update detection rules based on emerging **OWASP LLM Top 10** threats.

**END**