IIT KANPUR
POWERED BY:
HACK IITK
CYBERSECURITY CHALLENGE HACKATHON 2026

Local LLM Powered Policy Gap Analysis and Improvement Module

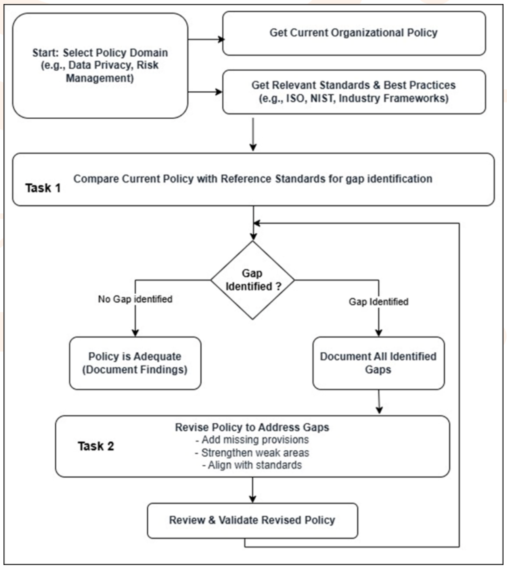
Background:

Organizational policies are critical governance documents, and their effectiveness depends on comprehensive, well-defined terms. While many organizations maintain policies for specific domains, such as Information Security Management System, Data privacy and security, Patch management, and Risk management etc., these policies often contain inadequate or incomplete provisions. To ensure robustness, existing policies must be benchmarked against industry standards and best practices to identify deficiencies.

Objective:

To identify gaps in the current policy by comparing it against relevant frameworks and standards, and revise the policy to address these identified shortcomings.

Flowchart of Proposed Module:


Start: Select Policy Domain (e.g., Data Privacy, Risk Management)
Get Current Organizational Policy
Get Relevant Standards & Best Practices (e.g., ISO, NIST, Industry Frameworks)
Task 1
Compare Current Policy with Reference Standards for gap identification
Gap Identified ?
No Gap identified
Gap Identified
Policy is Adequate (Document Findings)
Document All Identified Gaps
Task 2
Revise Policy to Address Gaps
- Add missing provisions
- Strengthen weak areas
- Align with standards
Review & Validate Revised Policy

Solution Requirements:

The solution must be implemented using a lightweight Large Language Model (LLM) capable of running entirely on a local machine. The system must operate completely offline without requiring an internet connection. No external APIs or cloud-based services should be utilized in the implementation.

Requirements:

- Lightweight LLM running locally
- Fully offline operation
- No external API dependencies

Technical Constraints:

1. Local deployment only - no cloud dependencies
2. Lightweight LLM optimized for local execution
3. Complete offline functionality
4. Zero external API integration

Deliverables

The deliverables for this work include the following components:

Code Implementation

A Python function or script that accepts a policy document as input and performs the following tasks:

A. Identifies gaps based on the *CIS MS-ISAC NIST Cybersecurity Framework Policy Template Guide (2024)*, available at:

> https://www.cisecurity.org/-/media/project/cisecurity/cisecurity/data/media/files/uploads/2024/08/cis-ms-isac-nist-cybersecurity-framework-policy-template-guide-2024.pdf

B. Revise the existing policy to address the identified gaps and suggests a key roadmap for improvement aligned with the NIST Cybersecurity Framework.

Documentation

Clear and comprehensive documentation explaining the following:

A. How to run the Python script or function.

B. Dependencies and installation instructions.

C. The logic and workflow of the entire code implementation.

D. Discussion of potential limitations and areas for future improvement.

Data Requirements

0.1 Organizational Policy Data (Test Data)

To validate the implementation, dummy organizational policies will be created covering the following cybersecurity processes:

- Information Security Management System (ISMS)
- Data Privacy and Security
- Patch Management
- Risk Management

These dummy policies will serve as test data, and the output of the gap analysis and policy revision process will be evaluated against them.

Policy Standards and Framework Templates (Reference Data)

The reference data for identifying gaps and aligning policies with recognized standards will be derived from the following policy template and framework guide:

> https://www.cisecurity.org/-/media/project/cisecurity/cisecurity/data/media/files/uploads/2024/08/cis-ms-isac-nist-cybersecurity-framework-policy-template-guide-2024.pdf

This document provides structured guidance based on the NIST Cybersecurity Framework and serves as the baseline for comparison and policy enhancement.