CAMILLE ARCHAMBAULT, McGill University

An Agentic Pipeline Combining GraphRAG and UMAP for Explainable Vulnerability Discovery in Low-Level Code.

Identifying the root cause of vulnerabilities in low-level code is difficult, time-consuming, and requires expert knowledge. Understanding the cause, not just the visible symptom, is essential for patching and analyzing security impact. However, low-level code provides little structural context: during compilation, programs are re-organized into many small blocks, and a vulnerability may appear in one location even though its true cause lies elsewhere. Existing tools typically highlight where the problem is detected but cannot trace the underlying source-sink chain that leads to it. With the rise of Large Language Models (LLMs), new opportunities emerge for automating vulnerability discovery while improving transparency in vulnerability analysis.

To address this challenge, we treat the binary and its low-level code as a searchable knowledge base that the LLM can query during analysis. However, because vulnerability causes span long chains across multiple functions, standard RAG is insufficient. We therefore turned to GraphRAG, which incorporates graph relationships between code elements but is computationally expensive on large graphs and still lacks a global semantic view of the program. Our pipeline therefore also leverages UMAP to organize code embeddings into a compact semantic space. This combination allows the agent to quickly identify relevant blocks of code before performing focused graph traversal, enabling more efficient discovery of source-sink paths and producing more interpretable explanations of low-level vulnerabilities.

This work is part of an ongoing master's thesis project and presents a research direction for explainable vulnerability discovery in assembly code, laying the foundation for future implementation and evaluation.