Project Plan: Automated Dynamic Analysis Signature Generation (Project 3)

Ikram Benfellah, Fadel Fatima Zahra

January 25, 2026

1 Project Information

- **Project Title:** Automated Dynamic Analysis Signature Generation
- **Team Members:** Ikram Benfellah , Fadel Fatima Zahra
- **GitHub Repository:** `https://github.com/ykram051/Automated-Dynamic-Analysis-Signature-Generati`

2 Methodology

High-level Approach:

1. Parse CAPEv2 JSON logs to extract behavioral patterns (API calls, network, file ops).
2. Use LLMs to identify and summarize key behaviors.
3. Automatically generate Python-based CAPEv2 signatures.
4. Map behaviors to MITRE ATT&CK techniques.
5. (Optional) Generate comprehensive, human-readable malware analysis reports.

System Architecture:

Technology Stack:

- Python 3.10+, PyTorch/Transformers (for LLMs), CAPEv2, MITRE ATT&CK framework, LaTeX (reporting)

3 Implementation Plan

Timeline and Milestones:

- **Feb 1-10:** Dataset acquisition and preprocessing
- **Feb 11-20:** Log parser and feature extraction
- **Feb 21-Mar 1:** LLM-based behavior analysis
- **Mar 2-10:** Signature generation and validation
- **Mar 11-15:** MITRE ATT&CK mapping
- **Mar 16-20:** Report generation, evaluation, and final write-up
- **Mar 22:** Final submission

Dependencies and Risks:

- LLM access (API limits, cost)
- Dataset size/quality
- CAPEv2 compatibility
- Mitigation: Early testing, fallback to manual feature extraction if needed

4 Research Component

4.1 Research Questions

1. What is the precision, recall, and F1-score of LLM-generated CAPEv2 behavioral signatures compared to manually crafted signatures on a held-out malware test set?
2. To what extent do LLM-generated signatures detect unseen variants from new malware families (i.e., generalization rate on out-of-family samples)?
3. How accurate is the automated mapping of behaviors to MITRE ATT&CK techniques (measured by correct technique assignments vs. expert annotation)?

4.2 Related Work

Implementations for Comparison:

- **Graph-Ensemble Methods:** No public code found; method described in detail for reproducibility.
- **DeepSign:** No official code, but method is reproducible from paper.
- **Automatic-Malware-Signature-Generation:** GitHub repository available.

Recent work applies deep learning and graph neural networks to automate malware signature generation and analysis. DeepSign and graph-ensemble methods focus on robust, invariant signature creation from behavioral logs, but do not target CAPEv2 or automate Python signature generation. Open-source tools provide automation but lack ATT&CK mapping and LLM integration. Our work uniquely automates CAPEv2 Python signature generation, ATT&CK mapping, and LLM-driven report synthesis for dynamic malware analysis.

Paper/Tool	Approach	Key Contribution	Our Difference
Graph-Ensemble Methods (2024)	Ensemble of GNNs and NLP for API call sequence analysis	Improves detection accuracy using graph models, automates signature generation with GNNExplainer, maps to MITRE ATT&CK	We use LLMs for CAPEv2 Python signature generation, not GNNs.
DeepSign (2017)	Deep learning (DBN, denoising autoencoders) for invariant malware signature generation	Achieves 98.6% accuracy on new malware variants; agnostic to input type (API calls, registry, etc.)	We automate CAPEv2 Python signatures and ATT&CK mapping with LLMs.
Automatic-Malware-Signature-Generation (GitHub)	Open-source tool for automated malware signature generation	Provides practical implementation and code for signature automation	We target CAPEv2 format and add ATT&CK mapping using LLMs.

Table 1: Related work comparison

4.3 Dataset Selection

- **Dataset:** AVAST-CTU CAPEv2 Dataset (`https://github.com/avast/avast-ctu-cape-dataset`)
- **Justification:** Large, diverse, public, includes CAPEv2 logs
- **Statistics:** >100,000 samples, multiple malware families
- **Preprocessing:** Filter for relevant behaviors, split into train/val/test
- **Split Strategy:** Chronological partitioning to avoid data leakage

5 Evaluation Plan

- **Metrics:** Precision, recall, F1-score (signature quality), coverage, ATT&CK mapping accuracy, efficiency (time savings)
- **Baselines:** Existing CAPEv2 signatures, manual analyst signatures, rule-based detection
- **Experimental Setup:** Evaluate on held-out test set, compare to baselines, statistical significance via paired t-test
- **Expected Outcomes:** LLM-generated signatures are competitive with manual, improve efficiency, and provide robust ATT&CK mapping