Will Young
Student ID: 924230057
Week 2 Report
2/6/2026

Hello Professor,

Below is the suggested timeline for this semester's project. Dates may become dynamic as we get going, however having a structured breakdown of the entire project into smaller goals should help keep me on track. In addition, I have created a github repo: https://github.com/Woyoung21/LastMile-Sec where I will be maintaining my code as well as my documentation.

Phase 1: Foundation & Data Ingestion (Feb 6 - Feb 27)
Focus: Setup environment, select models, start building the pipeline to handle raw data.

- Week 2 (Feb 6): Timeline development & Environment Setup
  - Create project timeline
  - Create github repo
  - Ensure IDE linked with proper dependencies
- Week 3 (Feb 13): Model Selection & Ingestion Pipeline Development
  - Review LLM models on Hugging Face (Mistral-7B vs Gemini) for the “Reporter” and “Mapping” roles
  - Build parsers for unstructured data (PDFs, Excel, CSV, PCAP)
  - Develop the normalization logic to convert raw excerpts into structured JSON packets
- Week 4 (Feb 20): Integrate the Reporter Model
  - Implement the “Reporter” model to take JSON packets and generate a single technical summary sentence
  - Test extracting key evidence while reducing “noisy” data

Phase 2: Analysis & Mapping (Feb 28 - Mar 20)
Focus: Connecting the summarized events to the MITRE ATT&CK framework.

- Week 5 (Feb 27):The “Mapper” Model Implementation
  - Locate public data set for fine tuning model on MITRE
  - Fine-tune the “Mapper” model using LoRA to better correlate technical sentences to MITRE ATT&CK IDs
  - Review Esposito’s thesis architecture to ensure correct implementation of the Reporter/Mapper split
- Week 6 (Mar 6):Integration Testing
  - Link Section 1 (Ingestion) and Section 2 (Mapping) into a unified workflow
  - Verify that a raw PDF report input successfully results in a MITRE ATT&CK ID output
- Week 7 (Mar 13): Knowledge Base Construction
  - Begin gathering vendor documentation (Meraki, AWS, CrowdStrike etc) and CIS benchmark for the RAG database

Phase 3: Synthesis & Remediation (Mar 21 - Apr 10)
Focus: Contextualizing the data for the specific client and generating fixes.

- Week 8 (Mar 20): RAG Implementation
    - Populate the RAG db with the vendor hardening docs
    - Develop logic to inject the client tech stack into the context window at the end of section 3
- Week 9 (Mar 27): Remediation generation
    - Build the LLm interaction that takes the ATT&CK ID + Client Context and retrieves vendor-tailored remediation steps
- Week 10 (Apr 3): Reliability & Hallucination Checks
    - Implement the “Self-RAG” check to verify guidance is based on real events and documentation, reducing hallucinations

Phase 4: Actionable Output & Final Polish (Apr 10 - May 8)
Focus: Formatting for L1/L2 engineers and final reporting.

- Week 11 (Apr 10): Actionable Output Generation
    - Refine the output to generate step-by-step or click-by-click UI walkthroughs (eg “Meraki dashboard -> Security…”)
    - Ensure tone is appropriate for L1/L2 engineers
- Week 12 (Apr 17): Human-in-the-Loop & Validation
    - Design the review interface or process where a human engineer validates the proposed remediation
    - Run full end-to-end test with real vulnerability reports
- Week 13 (Apr 24):Final Report & Documentation
    - Draft the final project report
    - Document the code and create the final presentation
- Week 14 (May 1): Buffer & Final Polish
    - Final code cleanup
    - Small buffer incase prior timeline need to be expanded