

## 문제정의서(연구개발계획서)

|                 |                               |
|-----------------|-------------------------------|
| Project<br>Name | 바이너리 프로그램에서 제어 구조를 식별하는 도구 개발 |
|-----------------|-------------------------------|

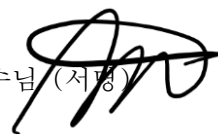
14 조

202002514 안상준

202202602 손예진

202202487 박혜연

지도교수: 조은선 교수님 (서명)



# Document Revision History

---

| REV# | DATE       | AFFECTED SECTION | AUTHOR |
|------|------------|------------------|--------|
| 1    | 2025/04/01 | 초안 작성            | 박혜연    |
| 2    | 2025/04/03 | 인터뷰 내용 추가        | 박혜연    |
|      |            |                  |        |
|      |            |                  |        |

# Table of Contents

---

|                             |   |
|-----------------------------|---|
| 1. 연구 개발의 필요성.....          | 5 |
| 2. 연구 개발의 목표 및 내용.....      | 5 |
| 3. 이해당사자 인터뷰/ 설문 인사이트 ..... | 7 |
| 4. 기대 효과 및 향후 확장 가능성 .....  | 8 |
| 5. 연구 개발의 추진전략 및 방법 .....   | 8 |
| 6. AI 도구 활용 정보.....         | 8 |
| 7. 참고문헌(REFERENCE) .....    | 9 |

# List of Figure

---

|    |                |   |
|----|----------------|---|
| 1. | 브레인스토밍 결과..... | 5 |
| 2. | 브레인스토밍 결과..... | 7 |

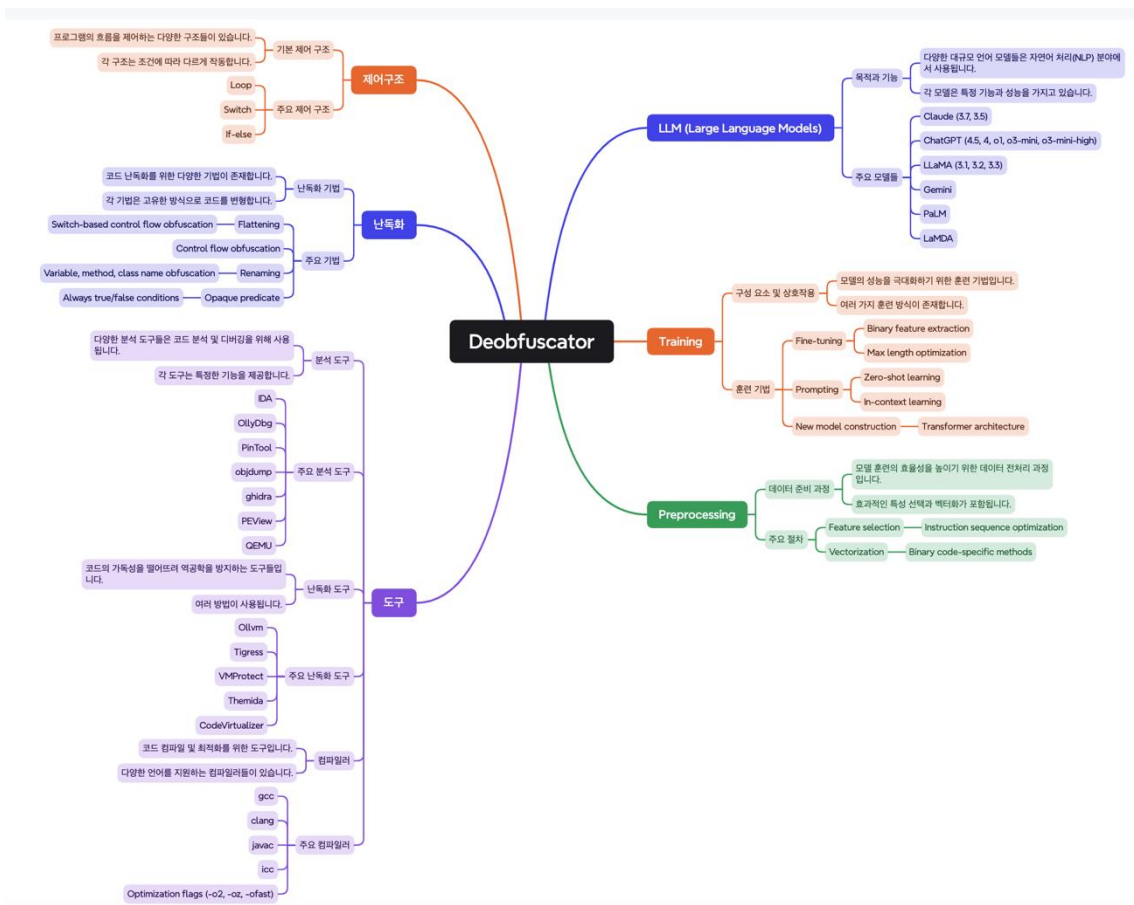
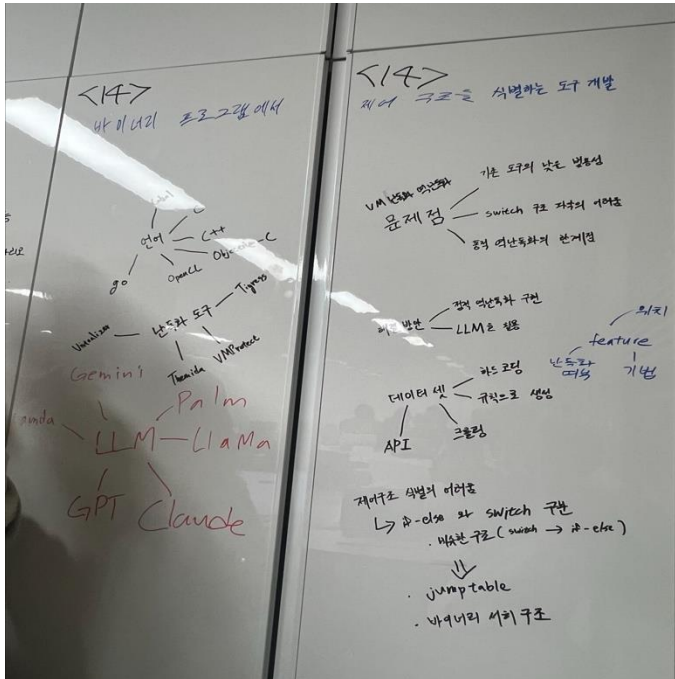
## 1. 연구 개발의 필요성

악성 코드를 빠르게 분석하는 것은 매우 중요하지만, 사람이 직접 코드를 분석하는 과정에는 많은 시간과 배경 지식이 필요하다. 이를 해결하기 위한 기존 연구에서 제안된 역난독화 도구는 특정한 난독화 도구나 기법에 최적화 되어있는 경우가 많다. LLM을 역난독화에 활용한다면 난독화 기법이나 도구에 영향을 받지 않고 분석에 필요한 시간도 확연히 줄어들 것이다. 현재까지 LLM을 역난독화에 사용하지 않은 이유는 신뢰성 문제와 데이터셋 문제가 가장 크다고 생각한다. 따라서 데이터셋을 직접 구축하고 이를 LLM 학습에 사용할 계획이다. 다양한 데이터셋이 확보된다면 이 도구는 플랫폼에 독립적으로 사용 가능할 것이다.

## 2. 연구 개발의 목표 및 내용

.LLM을 활용하여 가상화 난독화 된 코드를 역난독화 하는 것이 가장 큰 목적이다. 이를 통해 위/변조 된 프로그램이나 난독화 된 악성코드를 빠르게 탐지하여 피해를 막을 수 있다. 난독화 기법에는 flattening, opaque predicate, control flow, renaming 등을 고려하고 있다. 모델에 fine-tuning을 하거나 few-shot을 사용해 prompting을 하는 방식을 고려 중이다. 이 과정에서 최대한의 성능을 얻기 위해 데이터셋에 feature selection과 max length 조절이 필요하다. 벡터화의 경우에도 기존 자연어 처리 방식이 아니라 바이너리 코드에 맞는 벡터화 방식을 사용해야 한다.

다음은 효과적인 역난독화 방안을 고안하기 위한 브레인스토밍 결과다.



### 3. 이해당사자 인터뷰/ 설문 인사이드

2~30대 대학원생 두명을 대상으로 직접 인터뷰를 진행하였다.

#### 기존 역난독화 방식에 대한 불편한 점

- 기존 방식들은 특정 난독화 도구에 맞춰진 경우가 많아 약간의 변형이 있다면, 범용성이 낮다는 문제점이 있다.
- 일반적으로 난독화 된 데이터를 바로 사용할 수 없어 전처리 및 단순화 작업을 직접 수행해야한다는 문제점이 있다. 또한, 이 작업이 올바르게 수행되었는지 확인하는 것도 어렵다.

#### LLM을 사용하여 역난독화 하는 방식에 대해 어떻게 생각하는지/어떤 효과나 부작용을 예상하는지

- 빠른 속도로 효과적인 수행이 가능할 것이라고 기대한다.
- 기존 방식보다 더 포괄적인 역난독화 가능하지만, 수학이나 알고리즘으로 검증하지 못해 성능에 대한 신뢰도가 낮을 수 있다.
- LLM의 입력에는 제약이 존재하고, hallucination 현상을 우려한다.

#### 어느정도의 정확도가 보장되어야 사용할 의향이 있는지

- 50% 이상만 되어도 참고용으로 사용할 의향이 있다.
- 평가 기준에 따라 다르겠지만, 분석가들이 잘못된 정보에 시간을 낭비하지 않도록 FP와 FN 값이 낮아지게 해야할 것이다.

#### 추가적으로 제안하고싶은 방안

- 특정 구조의 유무보다는 어느 부분에 어떤 구조가 나타나는지 시각화 해주면 좋을 것 같다.
- 코드의 크기가 커져도 쉽게 사용할 수 있으면 유용할 것 같다.

인터뷰 내용을 바탕으로 정리해보자면, 난독화 방식의 여러 변형에 대응할 수 있는 도구가 필요하다. 또한, 이 모델의 성능을 검증하기 위한 평가 기준을 명확히 세우는 것이 중요해보이는데, FP와 FN을 고려하기 위해서 precision과 recall도 평가 기준에 포함하면 좋을 것 같다. 기존 방식에 필요한 전처리 작업은 알지 못했기 때문에 이에 대한 처리 방안과 결과를 시각화 하기 위한 학습 데이터 처리 방안에 대해서도 고려가 필요할 것 같다.

## 4. 기대 효과 및 향후 확장 가능성

데이터셋을 직접 구축하고 LLM 학습에 사용할 예정이다. 이는 LLM의 신뢰성뿐만 아니라 데이터셋의 사용 가능성도 입증되는 것이다. 데이터셋은 추후 다른 연구에도 사용될 수 있다. 현재 chat GPT, claude와 같은 LLM이 역난독화를 수행해주는 것은 하지만, 바이너리 코드에 특화되어 있지 않기에 정확하지 않은 결과가 많았다. 개발 예정인 이 도구는 바이너리 코드로 fine-tuning 하여 더 높은 정확도를 얻을 수 있을 것이다. 또한, 사람이 직접 하던 작업을 LLM을 사용하여 대체하기 때문에 코드 분석에 필요한 시간이 상당히 줄어들 것이다.

## 5. 연구 개발의 추진전략 및 방법

우선, 바이너리 코드에서의 switch, if-else, while 구조의 패턴을 확인해보며 난독화, 컴파일러, 프로그램 사용법 등 프로젝트에 필요한 기본 지식을 학습하는 기간을 가졌다. 현재는 난독화를 적용하고 LLM에 입력하여 코드의 구조를 식별하는지 확인하는 실험을 진행해보고 있다. 데이터셋의 경우에도 현재 github API를 사용해 코드를 가져오고 있는데, 이에 대한 처리도 추후 진행할 것이다. 데이터셋 전처리 후에 LLM에 직접 학습을 진행하여 실험해보고 성능을 개선해 나갈 예정이다.

## 6. AI 도구 활용 정보

해당사항 없음



## 7. 참고문헌(Reference)

- Mohseni, Seyedreza, et al. "Can LLMs Obfuscate Code? A Systematic Analysis of Large Language Models into Assembly Code Obfuscation." *arXiv preprint arXiv:2412.16135* (2024).
- Li, Xuezixiang, Yu Qu, and Heng Yin. "Palmtree: Learning an assembly language model for instruction embedding." *Proceedings of the 2021 ACM SIGSAC conference on computer and communications security*. 2021.
- 박성우, 박용수 "VMProtect의 역공학 방해 기능 분석 및 Pin을 이용한 우회 방안" 정보처리학회논문지. 컴퓨터 및 통신시스템 10.11 pp.297-304 (2021) : 297.