P R O J E C T D O C U M E N T

Usecase Specification Document

Project 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	2025/04/11	초안 작성	손예진
2	2025/04/12	1.2, 1.3, 1.7 수정	안상준

Table of Contents

List of Figure

Introduction

1.1. 연구 배경

악성코드는 발견되었을 때 빠르게 분석하는 것이 매우 중요하다. 그러나 가상화를 기반으로 난독화된 악성코드의 증가와 함께 이를 분석해내는 것이 점차 어려워지며, 빠르게 분석할 수 있는 도구가 필요성이 나타나고 있다.
최근 LLM이 다양한 분야에서 사용되고 있으나, 난독화/역난독화 등의 보안 분야에서는 다른 분야에 비해 활용도가 떨어지는 것으로 보인다. 이는 데이터셋의 부족과 LLM의 신뢰성 문제가 주된 원인으로, 해당 문제를 보완해냄으로써 LLM을 사용한 역난독화 도구를 개발해낸다면 가상화 기반의 난독화로 만들어진 악성코드의 분석에 활용할 수 있을 것으로 여겨진다.

1.2. 연구 목적

난독화는 코드의 가독성을 낮추는 것으로 난독화된 코드를 분석하여 그 원본을 추적하는 것이 역난독화이다. 난독화된 코드는 각 기법에 따라 특정적인 특정적인 패턴을 보인다.
본 연구에서는 바이너리 프로그램에서 제어 구조를 식별하는 도구 개발을 통해 가상화 난독화 코드를 역난독화 하는 연구에 지원할 계획이다. 특히, 가상화 난독화의 결과 형태로 나타나는 switch 구조에 대해 flattening, opaque predicate 등을 적용한 데이터셋을 구축하고 모델에 fine-tuning을 하거나 few-shot을 사용하여 prompting하여 생성한 LLM이 제어 구조를 분석해내고 판별할 수 있는 것을 목표로 한다.
이를 통해 악성코드에 대한 보안에 기여하고 연구 과정에서 얻어지는 LLM의 신뢰성 및 데이터의 사용 가능성에 대한 입증하는 기회로 만들고자 한다.

1.3. 연구 질문/ 가설

본 연구는 다음과 같은 연구 질문에 답하고자 한다:

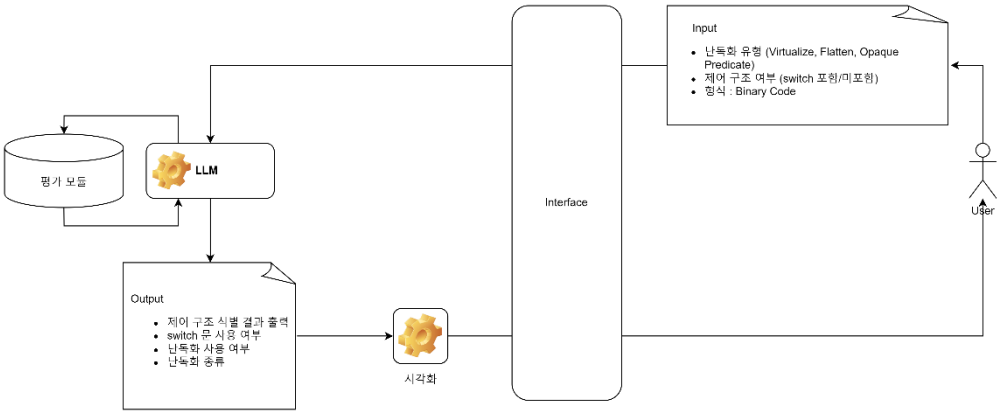
- RQ1.
  제어 구조(switch 문 등)에 대한 패턴 분석을 기반으로 구성된 데이터셋이 LLM 기반 역난독화 도구의 정확도 및 신뢰성 향상에 어떠한 영향을 미치는가?
- RQ2.
  LLM에 fine-tuning 혹은 few-shot prompting을 통해 구성된 역난독화 도구는 다양한 난독화 기법(flattening, opaque predicate 등)에 따라 분석 성능에 차이를 보이는가?

본 연구는 다음과 같은 가설을 설정할 수 있다:

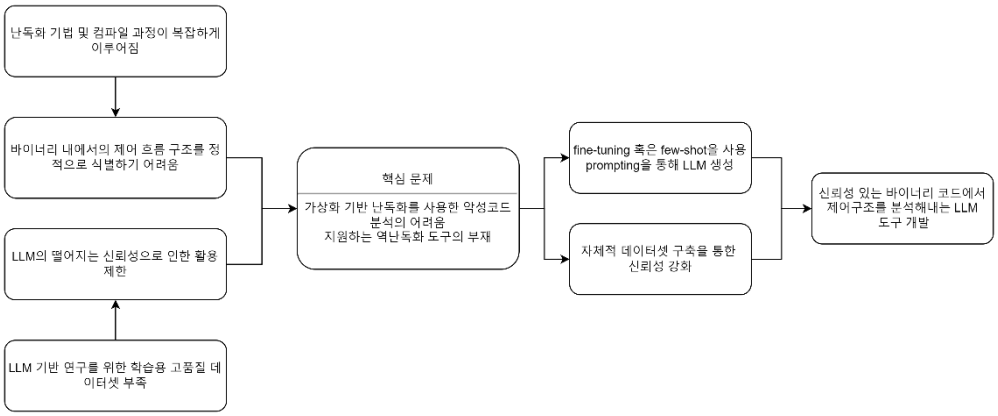
- H1.
  가상화 난독화된 코드의 구조인 fetch-decode-execute 구조를 식별 함으로써 코드의 원래 흐름과 구조를 분석하고 이는 역난독화에 큰 도움이 될 것으로 예상한다.
- H2.
  다양한 데이터셋을 활용하여 LLM에 fine-tuning 또는 few-shot prompting을 통해 학습시킨다면, 여러 난독화 기법에 대하여 유연하게 제어구조를 식별할 수 있을 것으로 기대한다.

Usecase Diagram

1.4. 소프트웨어의 사용 사례 Diagram



1.5. 문제 해결에 대한 사용 사례 Diagram



Usecase Specification

1.6. 소프트웨어 활용 사례

주요 Actor	User(사용자)
주요 기능 구성 요소	- LLM을 사용하여 Binary Code의 제어 구조(특히 switch 문)를 식별하여 역난독화 및 코드 분석 지원 - 난독화 유형에 관계없이 제어 흐름 정보를 추출할 수 있도록 학습된 LLM 활용
입/출력 데이터	입력 데이터 : • 난독화 유형: o Virtualize o Flatten o Opaque Predicate o 난독화 되지 않은 코드 • 제어 구조 여부: o switch 문이 포함된 코드 o switch 문이 포함되지 않은 코드 • 형식: Binary Code 출력 데이터 : • 제어 구조 식별 결과 o switch 문 사용 여부 o 난독화 사용 여부 o 난독화 종류
데이터 Flow	1) 사용자가 분석 대상 바이너리 코드를 입력 2) LLM은 코드의 제어 흐름 패턴을 분석하여 결과를 출력 3) 출력 후처리 모듈이 결과를 정리하여 시각화
외부 시스템 연계	평가 모듈 : ground-truth 데이터와 LLM 예측 결과를 비교하여 정확도 / 정밀도 / 재현율 등을 측정

1.7. 문제 해결에 대한 사용 사례

핵심 문제	가상화 기반 난독화를 사용한 악성코드의 분석의 어려움과 이를 지원하는 역난독화 도구의 부재
직접 요인	바이너리 코드 내에서의 제어 흐름 구조를 정적으로 식별하기 어려움 LLM의 떨어지는 신뢰성으로 인한 활용 제한
간접 요인	새로운 난독화 기법의 개발로 인해 대응하기가 어려움 LLM 기반 연구를 위한 학습용 고품질 데이터셋 부족
활용 맥락	악성코드 분석 환경: 분석가가 난독화된 바이너리 내의 제어 구조를 빠르게 파악하여 추적 가능하게 함 자동화 시스템: LLM 기반 제어 흐름 식별 도구를 통해 분석 효율 향상 및 탐지 자동화 구현 향후 데이터셋의 활용: 제어 구조 데이터셋과 도구를 활용한 학습 및 역난독화 연구 가능

2.AI 도구 활용 정보

사용 도구	GPT-4-turbo
사용 목적	연구 질문 작성, 데이터 Flow 작성, 활용 맥락 작성
프롬프트	● 연구 질문에 대한 예시를 작성해줘 ● 입력 데이터와 출력 데이터 등을 바탕으로 데이터 Flow 어떤식으로 이루어질지 작성해줘 ● 우리 연구의 결과가 어떤 맥락에서 활용될 수 있는지 작성해줘
반영 위치	1. 연구 질문/가설 (p.6) 2. 소프트웨어 활용사례 (p.8) 3. 문제 해결에 대한 활용 사례 (p.9)
수작업 수정	있음(논리 보강, 사례 교체 등 Gpt가 작성한 초안에 대한 수정)