Swify

바이너리 프로그램에서 제어구조를 식별하는 도구개발

2025.05.26
202002514 안상준
202202602 손예진
202202487 박혜연
조은선 교수님

Contents

01 프로젝트 개요 - 배경/문제/필요성

배경

- 사이버 위협의 증가로 악성코드 분석의 중요성 대두됨.
- 악성코드는 대부분 다양한 난독화 기법이 적용되어 분석이 어려움.
- 가상화 난독화는 일반적인 분석 도구로는 분석의 한계가 있음.

문제

- 가상화 난독화는 난독화 과정에서 loop-switch 형태로 추상화 된다.
- 난독화 된 코드에서 switch를 탐지하는 것은 매우 까다로움.

필요성

- 효율적이고 범용적인 역난독화 도구의 필요성이 커지고 있음.
- LLM의 코드 및 패턴 능력을 활용하여 자동화된 악성코드 분석 시스템 개발.

01 프로젝트 개요 - 팀원소개

이름	역할	주요 기여
안상준	데이터셋 생성 백엔드 개발	LLM 학습에 필요한 데이터셋 생성 웹서비스의 백엔드 개발
박혜연	바이너리 코드 분석 AI 실험	바이너리 프로그램에저 제어구조를 분석 기존 상용 LLM, Bert 모델 실험 및 비교
손예진	데이터 난독화 프론트엔드 개발	데이터셋을 난독화 하여 라벨링 웹서비스의 프론트엔드 개발

02 사용자 분석 – 설문 인사이트

Q1. 기존 역난독화 방식에 대한 불편한 점이 있습니까?

기존 방식들은 특정 난독화 도구에 맞춰진 경우가 있어 범용성이 낮다.
난독화 데이터의 전처리 및 단순화 작업이 필요하다.

Q2. LLM을 사용하여 역난독화 하는 방식에 대해 어떻게 생각하시나요?

기존 방식에 비해 장점이 존재하지만 검증이 필요하다.
LLM 입력에는 제약이 존재하고, Hallucination 현상의 우려

Q3. 어느 정도 정확도가 보장되어야 사용할 의향이 있나요?

50% 이상만 되어도 참고용으로 사용할 의향이 있음.
평가 기준에 따라 다르겠지만, FP와 FN 값이 낮아야 함.

인터뷰 INSIGHT

난독화 방식의 여러 변형에 대응할 수 있는 도구 필요

명확한 평가 기준 필요
FP, FN 을 고려하기 위한 percision, recall을 포함

결과를 시각화 하기 위한 LLM 학습에 사용할 데이터 전처리 방안 고려

03 핵심 아이디어

제안 방법

가상화 난독화된 코드는
fetch-decode-execute 구조를 가짐

```
while (1) {
  switch (*(_TIG_VZ_GXCt_1_main_$pc[0])) {
  case _TIG_VZ_GXCt_1_main_load_unsigned_long_long$left_STA_
  (_TIG_VZ_GXCt_1_main_$pc[0]) ++;
  (_TIG_VZ_GXCt_1_main_$sp[0] + 0)->_unsigned_long_long = *(
  break;
  case _TIG_VZ_GXCt_1_main_store_void_star$right_STA_0$left_
  (_TIG_VZ_GXCt_1_main_$pc[0]) ++;
  *((void **)(_TIG_VZ_GXCt_1_main_$sp[0] + -1)->_void_star)
  _TIG_VZ_GXCt_1_main_$sp[0] += -2;
  break;
  case _TIG_VZ_GXCt_1_main_store_unsigned_long_long$right_ST
  (_TIG_VZ_GXCt_1_main_$pc[0]) ++;
  *((unsigned long long *)(_TIG_VZ_GXCt_1_main_$sp[0] + -1)-
  _TIG_VZ_GXCt_1_main_$sp[0] += -2;
  break;
  case _TIG_VZ_GXCt_1_main_load_int$left_STA_0$result_STA_0:
  (_TIG_VZ_GXCt_1_main_$pc[0]) ++;
  (_TIG_VZ_GXCt_1_main_$sp[0] + 0)->_int = *((int *)(_TIG_VZ
  break;
  case _TIG_VZ_GXCt_1_main_store_int$right_STA_0$left_STA_1:
  (_TIG_VZ_GXCt_1_main_$pc[0]) ++;
  *((int *)(_TIG_VZ_GXCt_1_main_$sp[0] + -1)->_void_star) =
  _TIG_VZ_GXCt_1_main_$sp[0] += -2;
```

LLM

바이너리에서
switch 여부 탐지
난독화 기법 탐지

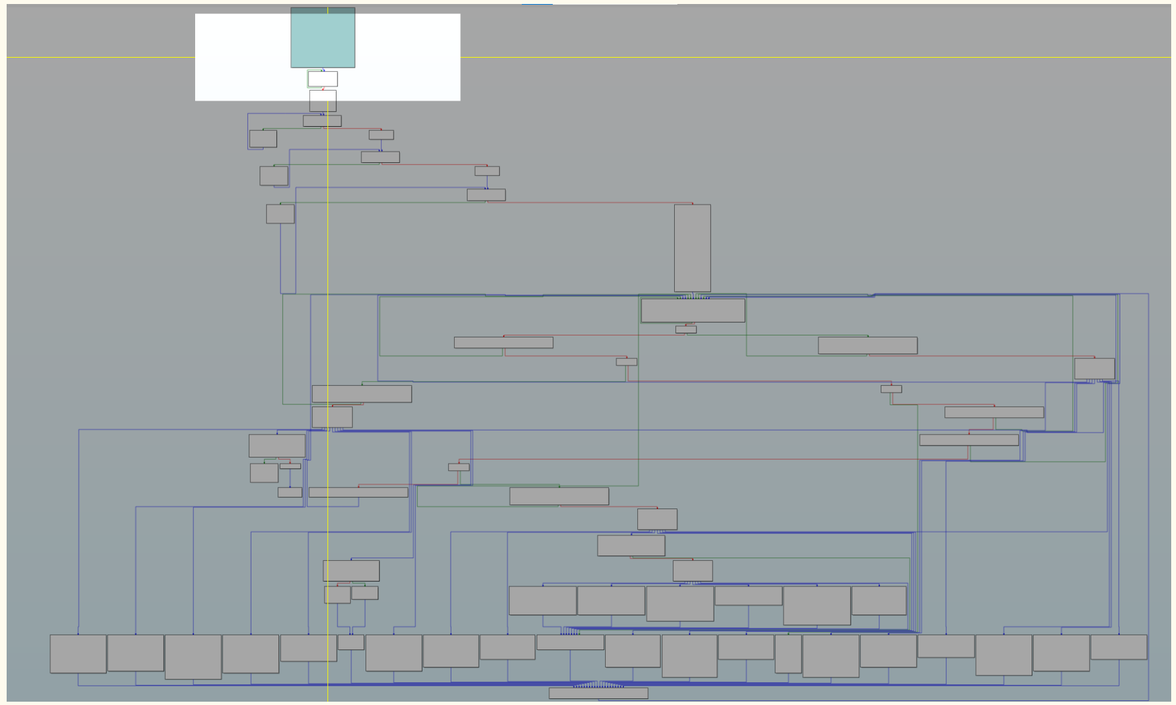
```
LFB16:
    .cfi_startproc
    pushl   %ebp
    .cfi_def_cfa_offset 8
    .cfi_offset 5, -8
    movl    %esp, %ebp
    .cfi_def_cfa_register 5
    andl    $-16, %esp
    subl    $32, %esp
    call    ___main
    movl    $LC6, (%esp)
    call    _printf
    leal    28(%esp), %eax
    movl    %eax, 4(%esp)
    movl    $LC7, (%esp)
    call    _scanf
    movl    28(%esp), %eax
    cmpl    $4, %eax
    ja  L8
    movl    L10(,%eax,4), %eax
    jmp *%eax
    .section .rdata,"dr"
    .align 4
```

03 핵심 아이디어

기존 방식

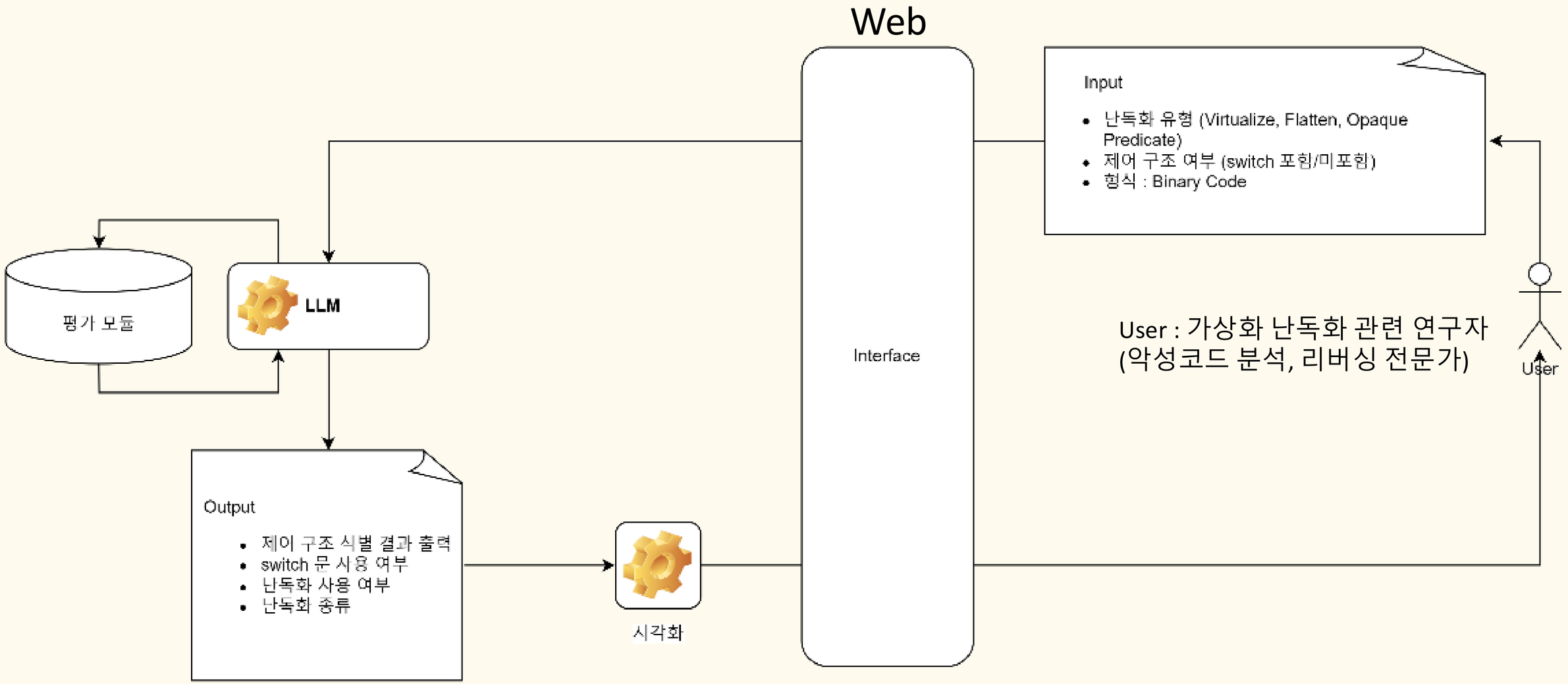
Reversing
정적/동적 분석도구를 사용

수백 개의 블록이 얽혀 있어 분석 난이도 ↑

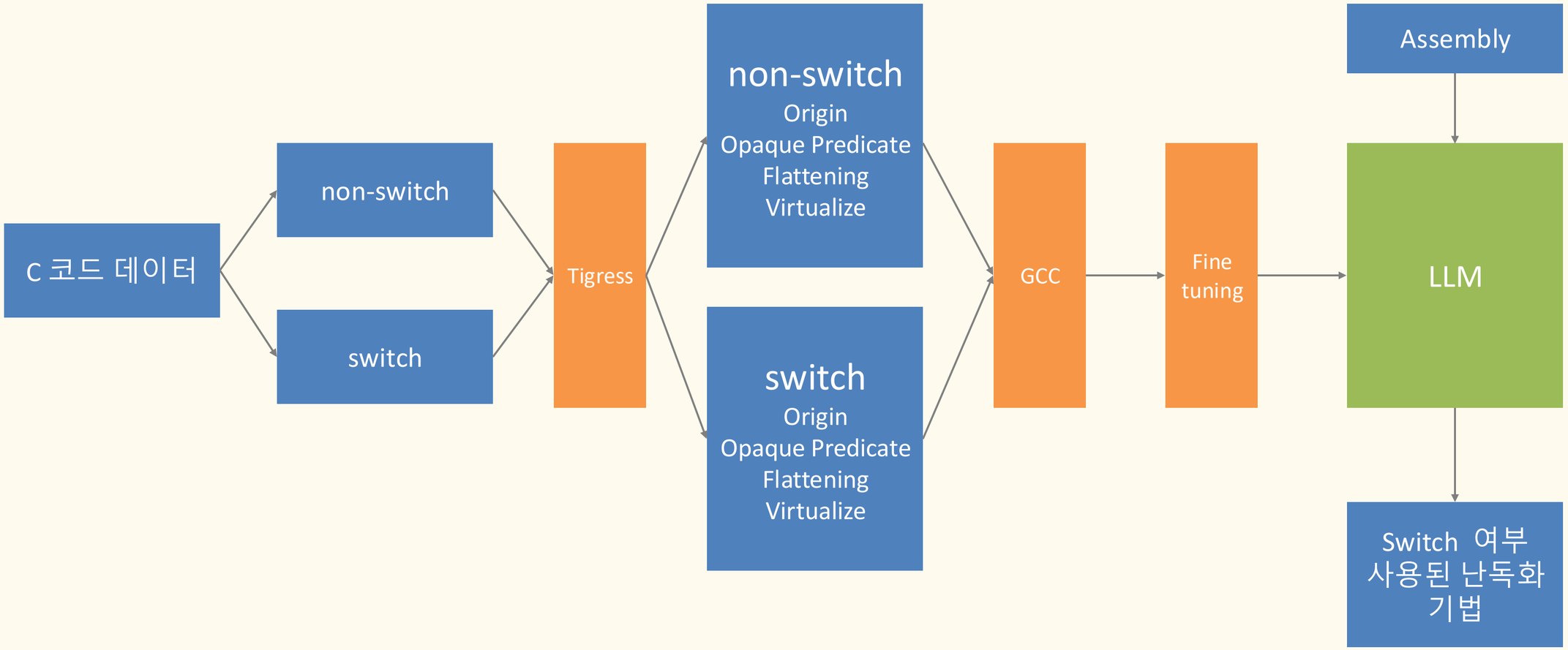


- **수작업**으로 분석
- **난독화 기법**에 따라 난이도 증가
- **분석 언어**에 제한

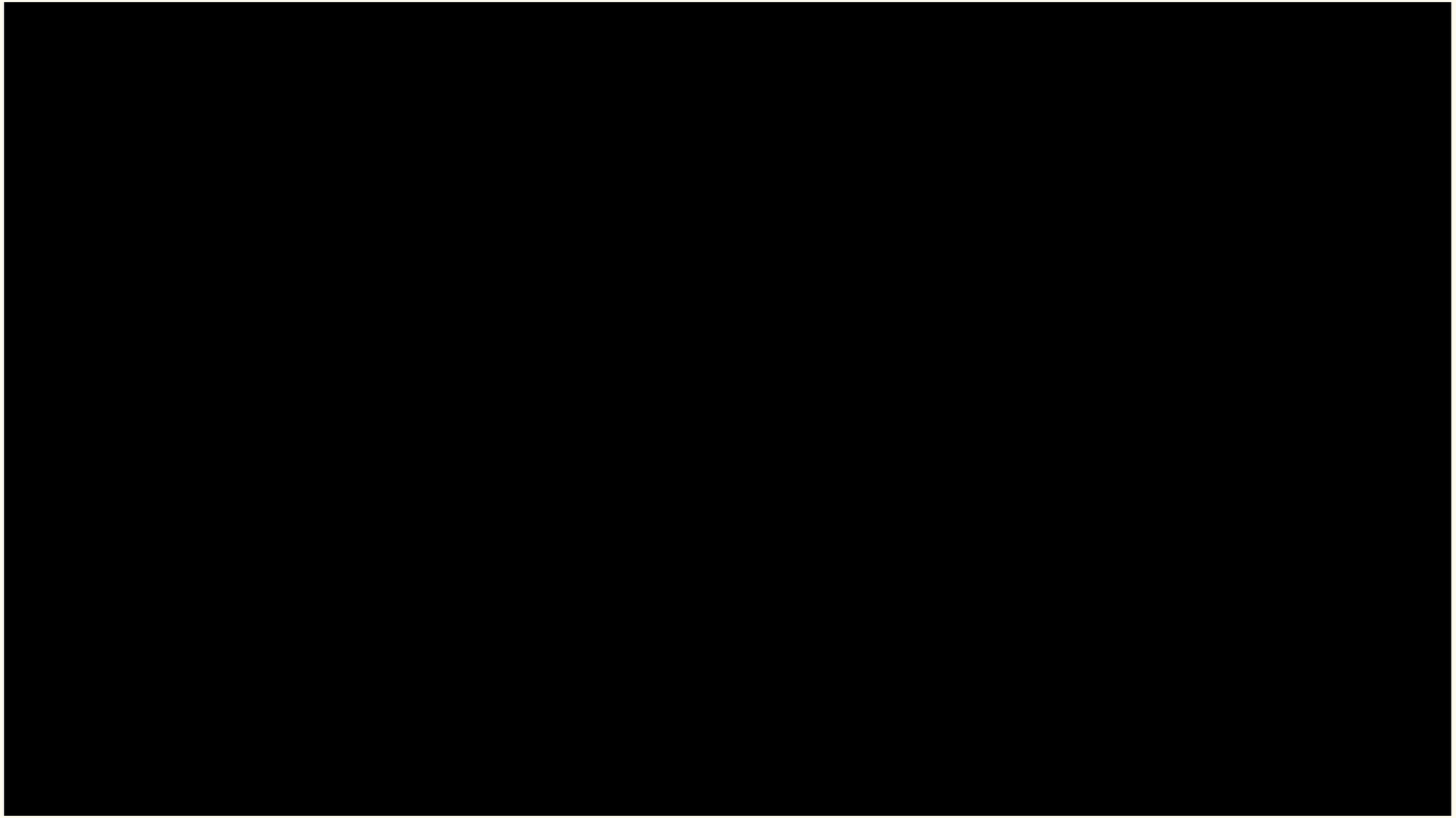
04 데모 – 핵심 유스케이스



04 데모 – 시퀀스 다이어그램



05 테스트 – 시제품 결과



05 테스트 – 성능 평가

실험 결과

평가 지표 / 난독화 기법	정확도 (Accuracy)	정밀도 (Percision)	재현율 (Recall)	F1 Score
Non	95.59%	1	0.93	0.96
Flattening	98.74%	0.97	1	0.98
Opaque Predicate	88.23%	0	0	0

06 추가 계획 기대 효과

추가 계획

- 난독화 기법을 추가 학습시켜, 탐지하는 난독화 기법 확장
- 제어구조의 위치를 마스킹 하는 기능 추가
- 두 개 이상의 난독화가 적용된 경우 인식 가능한 모델 학습

기대 효과

- 다양한 난독화 기법에 대해 자동화된 제어구조 탐지가 가능해짐
- 가상화 난독화 역난독화 연구에 기여
- LLM 기반 보안 분석 모델의 효과를 실험적으로 검증

DEOBFUSCATOR

감사합니다

컴퓨터융합학부 202002514 안상준

인공지능학과　202202487 박혜연

컴퓨터융합학부 202202602 손예진