LLM을 통한 취약점 자동 익스플로잇 도구 개발

2025학년도 2학기
SW 캡스톤디자인 경진대회

팀 명 AEG

지도교수 김윤경

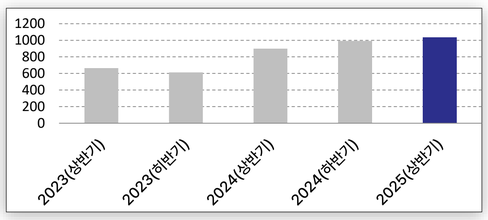
팀 원 조서윤(IT지능정보공학과, 3학년), 박채우(IT정보공학과, 4학년), 전승혁(컴퓨터공학과, 4학년), 이혜원(IT지능정보공학과, 4학년)

산업체 해커스페이스(주)

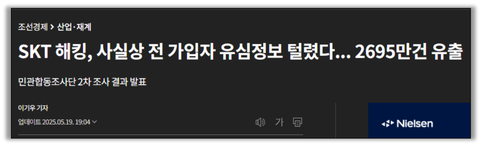
개발 동기 및 목적

현황 및 문제 인식

2025년 상반기 국내 사이버위협 동향



국내 보안 침해사고



- 최근 국내 개인정보 유출, 서비스 마비 등 보안 침해사고 지속적으로 증가
- 기존 보안 체계만으로는 진화하는 공격에 효과적 대응 어려움
- LLM의 추론 능력을 활용하여 취약점 탐지부터 익스플로잇 생성, 검증, 패치 제안까지 자동화된 보안 분석 파이프라인을 구축

자동화된 보안 파이프라인 구축

- 기존 보안 시스템이 진화하는 공격에 대응하지 못하는 한계 개선
- LLM의 추론 능력을 활용해 취약점 탐지부터 익스플로잇 생성까지 자동화
- 익스플로잇 검증 및 패치 제안까지 이어지는 보안 분석 파이프라인을 구축

주요 기술

Vulnerablility Analyzer

- CWE 유형과 취약 지점을 자동 식별
- 익스플로잇 생성을 위한 핵심 조건 추출

libFuzzer

- 취약 지점을 기준으로 다양한 입력을 자동 탐색
- 반복 실행을 통해 crash를 발견하고 기록

Exploit Generator

- Analyzer·Fuzzer 정보 기반으로 Exploit 자동 생성
- crash 원인을 분석해 payload 구조를 구성

PoV Verifier

- 생성된 exploit을 실행해 crash 재현 여부 확인
- 취약점 존재의 신뢰성을 검증

개발 내용

Automatic Exploit Generation

1. Preprocessing: 불필요한 주석, 중복 코드, 테스트 파일 제거 및 취약 함수 중심의 코드 요약
2. Vulnerability Analysis: 전처리된 코드를 기반으로 CWE 유형, 취약 함수, 트리거 조건을 JSON 형태로 추출
3. Fuzzing: 자동 생성된 harness 및 seed를 활용하여 실제 crash 수집
4. Exploit Generation: 분석 결과를 활용하여 LLM이 Exploit 코드를 자동 생성하도록 설계
5. PoV Verification: 생성된 Exploit 코드를 실제 실행하여 공격 성공 여부 검증
6. Patch Advisor: 안전한 코드 수정안 및 패치 권고문 자동 생성

결과 및 분석

기술적 성과

- 취약 코드에 대한 자동 분석과 PoC 검증수행 및 다양한 서비스 환경에서도 적용 가능한 범용 구조 확보
- CWE 기반 분석과 LLM 모듈을 결합하여 새로운 취약점 유형과 공격 기법에도 손쉽게 확장할 수 있는 유연성 확인
- 전체 절차가 자동화되면서 개발·보안팀의 triage 비용과 시간이 크게 절감되는 운영 효율 향상 효과 확인

기대효과