

\$ whoami

자기 소개



안녕하세요.

문제를 **주도적으로 찾고,**
주어진 문제를 끈기 있게
해결하는 김동건 입니다!

보유 기술

- C / C++ / Python / HTML/CSS/JS / Java/Kotlin
- x86, ARM64, MIPS64, s390x
- docker, Git

학력 및 이력

- 부경대학교 컴퓨터공학전공
(2018.03 ~ 2025.02 졸업 예정)
- KITRI BoB 취약점 분석 12기
(컴파일러 최적화 취약점 분석 PM)
- codeinject
도구 프로젝트 및 세미나 발표
- CTF 대회 출전
(UofTCTF 2024, Insomni'hack ,
hspace CTF)

GitHub Link(gbdngb12)

\$ Compiler Optimization Vulnerability Analysis

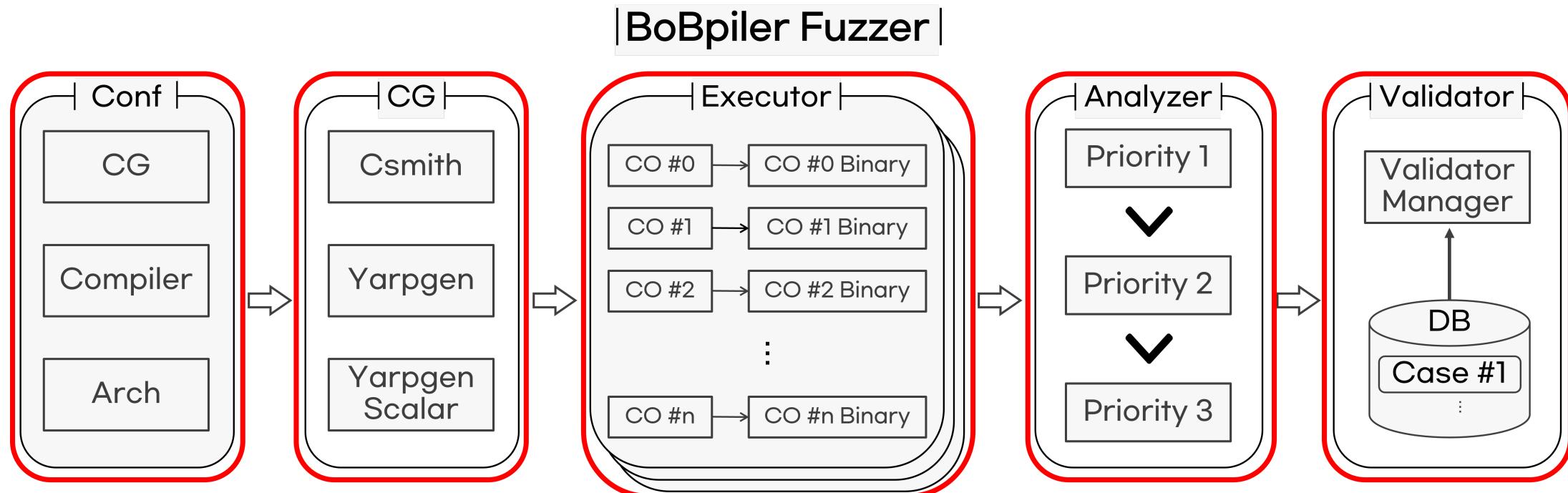
컴파일러 최적화 취약점 분석 프로젝트 요약

항목	내용
프로젝트 기간	2023.09 ~ 2023.12 (4개월)
진행 배경	컴파일러 최적화 과정은 매우 복잡 , 의도하지 않은 취약점 분석
역할	PM, 프로젝트 총괄 및 발표 (총 6명)
기술적 핵심 내용	컴파일러 Fuzzer 개발 및 CodeQL, 최적화 버그 취약점 연계
프로젝트 성과	컴파일러 버그 31건, 학술대회 논문 2개, Stealth Backdoor 시나리오 연구, PoC 3 개 개발, CodeQL Query 9개 개발
프로젝트 링크	GitHub

\$ Compiler Optimization Vulnerability Analysis

프로젝트 핵심 내용 : Fuzzer

*CG: Code Generator, *Arch: Architecture, *CO: Compile Options



CG에서 생성된 코드가 **Executor**에서 컴파일 및 실행, **Analyzer**, **Validator**에서 버그 탐지

\$ Compiler Optimization Vulnerability Analysis

프로젝트 성과 : Conference Paper Submission, Vulnerability PoC Using Compiler Bugs, CodeQL Query

학술대회 논문 투고, 컴파일러 최적화 버그가 취약점으로 연결 될 수 있음을 증명하는 PoC 개발, 컴파일러 최적화 버그를 탐지하는 CodeQL Query 개발

한국컴퓨터종합학술대회 (KSC)
“컴파일러최적화버그자동탐지체계”



김동건 포트 폴리오

한국정보보호학회 (KIISC)

“차분 퍼징을 활용한
아키텍처별 컴파일러 최적화 버그 탐지”



PoC 3개 개발

PoC1: Stack Buffer Overflow

PoC2: Use After Free

PoC3: Heap Overflow

CodeQL Query

```
bit.field.ql
```

/**
 * @name Bitwise AND operation between long long int and int types
 * @description finds instances where a bitwise AND operation is performed
 * between long long int and int types, which can result in undefined behavior.

Github : gbdngb12

\$ Compiler Optimization Vulnerability Analysis

프로젝트 성과 : Stealth Backdoor Scenario

악의적인 공격자가 컴파일러 최적화 버그를 이용해서 오픈소스에 코드를 삽입할 경우
특정한 컴파일러, 아키텍쳐, 최적화 옵션에서 취약점을 유발하는 시나리오 연구

The screenshot shows a GitHub repository page for 'Stealth Backdoor Scenario 1: OpenSSH'. The page title is 'Stealth Backdoor Scenario 1: OpenSSH'. Below the title is a 'Description' section with the following bullet points:

- Type: Password Authentication Bypass
- OS: Microsoft Windows 11 23H2
- Compiler: MSVC CL (ARM64)
- Compiler Version: 19.38.33133 (ARM64)
- Compiler Options: Problematic: /O1, /O2, /Ox Normal: /Od, /Ot
- Bug Behavior:
 - 해당 시나리오는 컴파일러 최적화 옵션에 따라 잘못된 연산을 수행하는 버그로 인해, OpenSSH 사용자 비밀번호 인증 결과를 우회하는 시나리오입니다.
 - Incorrect handling of authentication logic under specific compiler optimizations leads to a stealthy backdoor.
- Reported Bug: [Link to the reported bug](#)
- Open Source Repository: [OpenSSH GitHub Repository](#)

At the bottom of the page, there is a terminal window showing command-line output related to the exploit development for OpenSSH.

The screenshot shows a GitHub repository page for 'Stealth Backdoor Scenario 2: libxml2'. The page title is 'Stealth Backdoor Scenario 2: libxml2'. Below the title is a 'Description' section with the following bullet points:

- Type: Return Address Overwrite (Potential for Remote Code Execution)
- OS: Microsoft Windows 11 23H2
- Compiler: MSVC CL (ARM64)
- Compiler Version: 19.38.33133 (ARM64)
- Compiler Options: Problematic: /O1, /O2, /Ox Normal: /Od, /Ot
- Bug Behavior:
 - 해당 시나리오는 컴파일러 최적화 옵션에 따라 잘못된 연산을 수행하는 버그로 인해 libxml2의 함수 호출 스택에서 return address를 덮을 수 있습니다. 이 버그는 잠재적으로 원격 코드 실행(RCE)이 가능한 취약점으로 발전할 수 있으며, 이를 통해 스텔스 백도어가 구현될 수 있습니다.
 - This scenario exploits a bug in libxml2 due to incorrect operations under specific compiler optimizations, allowing the overwriting of the return address in the function call stack. This bug could potentially lead to Remote Code Execution (RCE) and serves as a basis for implementing a stealth backdoor.
- Reported Bug: [Link to the reported bug](#)
- Open Source Repository: [libxml2 GitLab Repository](#)

At the bottom of the page, there is a terminal window showing command-line output related to the exploit development for libxml2.

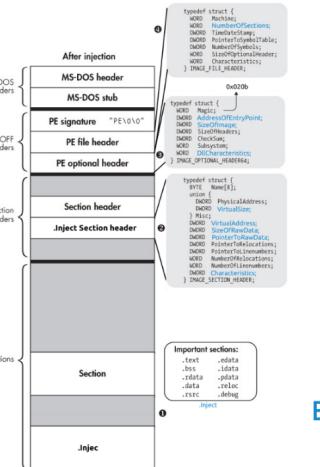
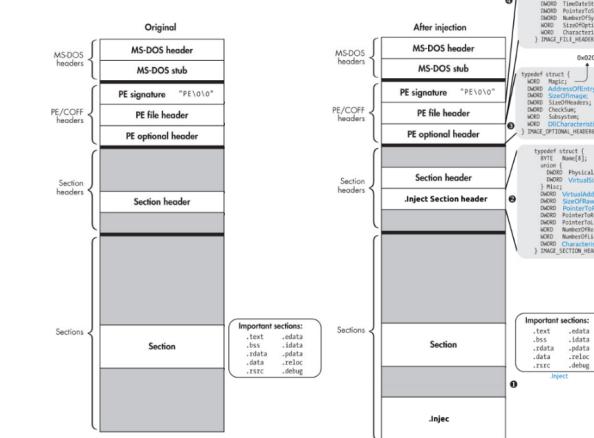
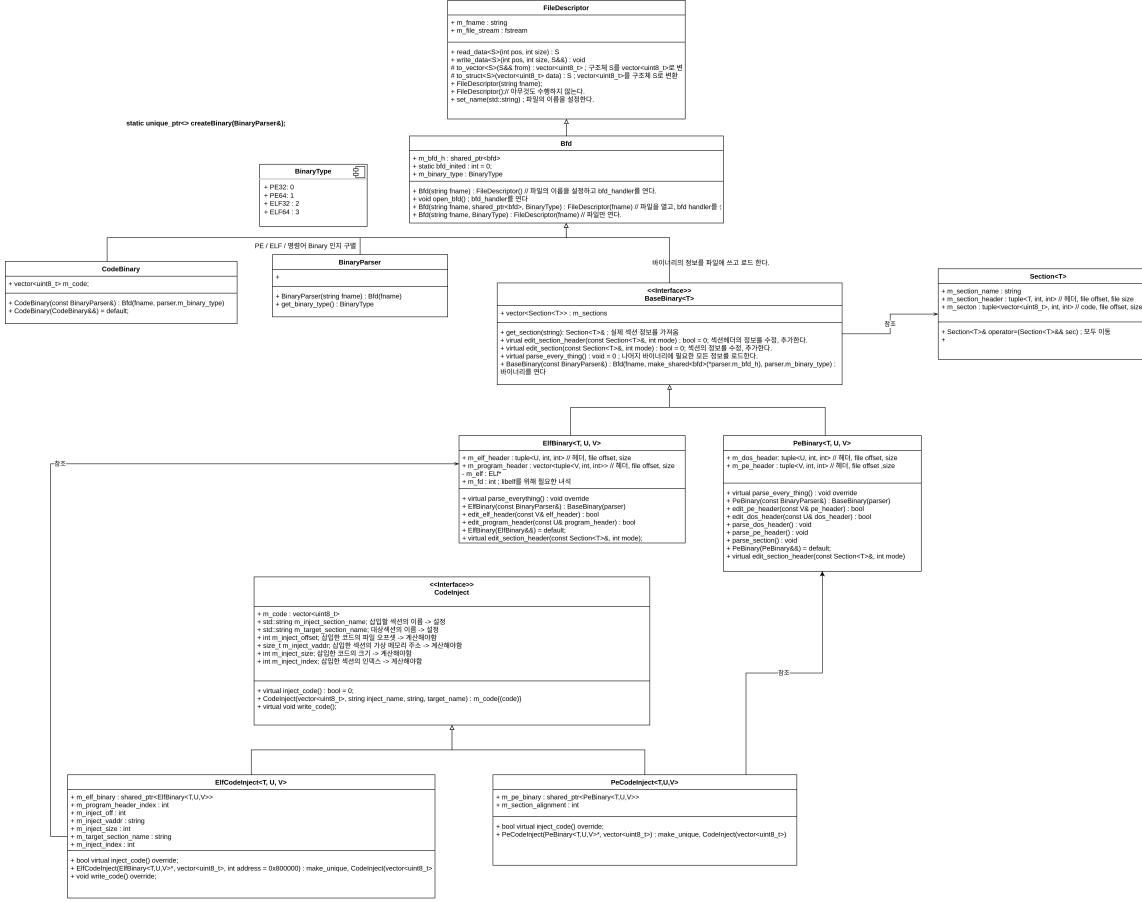
\$ Development codeinject Tool

codeinject 도구 개발 프로젝트 요약

항목	내용
프로젝트 기간	2023.05 ~ 2023.06 (1개월)
진행 배경	ELF, PE, 객체지향 C++ 학습 및 악성코드 도구 개발
역할	개인 프로젝트
기술적 핵심 내용	ELF, PE 구조, C++, Linux/Windows Loader의 역할, 클래스 다이어그램
프로젝트 성과	codeinject 도구, 부산대-부경대 세미나 발표
프로젝트 링크	GitHub

\$ Development codeinject Tool

프로젝트 핵심 내용 : Class Diagram, Principle Study



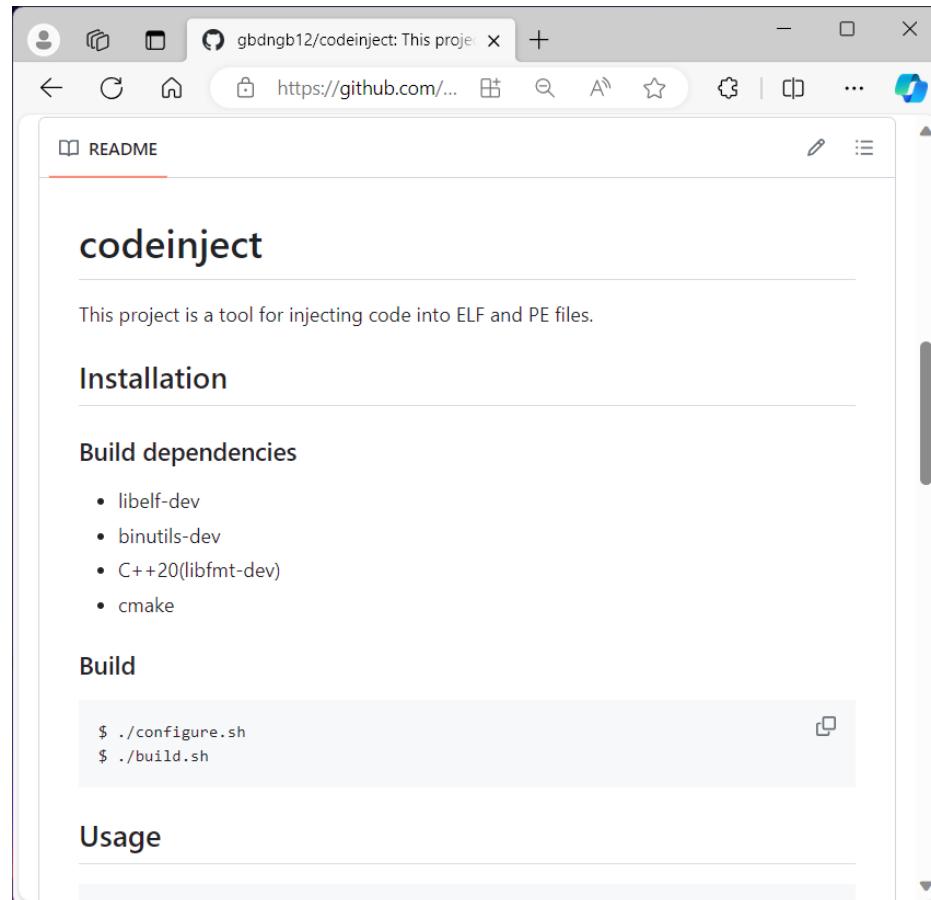
1. 코드를 파일의 끝에 삽입한다.
2. Section Header의 정보를 추가한다.
3. PE Header의 정보, Entry Point를 수정한다.
4. Section수를 수정한다.

결론

Entry Point를 탈취해 원하는 코드를 수행하고 원래의 제어 흐름으로 복귀한다.

\$ Development codeinject Tool

프로젝트 성과 : codeinject Tool, Seminar Presentation



지속적인 CTF 대회 출전 : UofCTF 2024, Insomni'hack, hspace CTF

Participated in CTF events

2024 2023 2021 2020 2019 2017 2015 2014 2013 2012 2011

Overall rating place: 393 with 5.340 pts in 2024

Place	Event	CTF points	Rating points
915	Real World CTF 6th	32.0000	1.460
259	Insomni'hack teaser 2024	4.0000	0.178
183	UofCTF 2024	2057.0000	3.701

올해에 열린 CTF대회에 출전함으로써
pwnable 기술을 공부하며, 취약점을 찾고
exploit할 준비를 하고 있습니다 !



dong_

김동건

1348 | Silver II

내 프로필 수정

워게임

시스템해킹	637
리버싱	0
웹해킹	90
암호학	0

Future Plan

1. Study Binary Fuzzing

바이너리 Fuzzing에 대한 공부 및 Fuzzing 진행

2. Analysis 1-Day Vulnerability

기존 취약점 1-Day 분석 진행

3. Find Real World Major Program Vulnerability

리얼 월드에서 메이저 프로그램의 취약점 찾기

김동건

Email : gbdngb12@naver.com

GitHub : <https://github.com/gbdngb12>

꾸준하게 하자.