Tracelet-Based Code Search in Executables

PLDI 2014

Yaniv David, Eran Yahav

김영철

2016. 3. 25.

Abstract

- goal : 정적으로 유사한 함수를 찾는 것
- 함수를 tracelets으로 분해하여 접근
  *tracelets : continous, short, partial traces of an execution
- low-level compiler transfomation의 tracelet 유사도 측정
  ➔
- 100만개가 넘는 바이너리 함수들에 적용
- n-grams, graphlets 기반의 방법들과 비교
  ➔ tracelets 매칭 방법이 더 좋은 것을 보여줌

Introduction

- 유명한 라이브러리에서 많은 취약한 부분이 발견됨

- 취약한 함수의 코드가 다양한 방법으로 포함될 수 있다.
  ➔ 효과적으로 탐지하는 방법이 없음
  ➔ 실행파일 내에서 효과적인 탐지 수단을 제공하는 것이 목표
  ➔ 정확한 매칭보다는 수정된 버전을 찾을 수 있는 유사성의 개념 정의

Introduction

- Existing Techniques

  (1) k-gram : syntactic or structural similarity only
  (2) k-gram & graphlets : 구조적 매칭을 위해서 graphlet 결합
  *graphlets : CFG의 모양이 같지 않은 subgraphs
  ➔ 유사도에 대한 신뢰성이 문제

  (3) Data-driven equivalence checking : 동적분석
  (4) Abstract semantic difference : static semantic-based

Introduction

- Tracelet-based matching

  (1) Tracelet decomposition
  함수의 CFG를 tracelet으로 분해

Introduction

- Tracelet-based matching
  (2) Tracelet similarity by rewriting