Comprehensive Evaluation of Cross Translation Unit Symbolic Execution

Endre Fülöp, Norbert Pataki

Department of Programming Languages and Compilers,
Eötvös Loránd University
gamesh411@gmail.com,patakino@elte.hu

Abstract

Static analysis is a great approach to find bugs and code smells. Some of the errors span across multiple translation units. Symbolic Execution is a major static analysis technique. Symbols are used to represent unknown values (e.g. user input), and symbolic calculations are carried out on them. Clang Static Analyzer (SA) is an open source symbolic execution engine for C/C++/Objective-C. SA had not support cross translation unit analysis, but now it is improved in this way.

In this paper, we evaluate the cross translation unit symbolic execution in a comprehensive way. Different caching methods, different approaches are considered. We compare the analysis of many open source projects. The aim is an optimal configuration for the tool.

Keywords: symbolic execution, cross translation unit, Clang

MSC: 68N15 Programming languages