Efficient and Safe Control Flow Recovery Using a Restricted Intermediate Language

Submitted by grigby1 on Thu, 09/26/2019 - 10:24am

Title	Efficient and Safe Control Flow Recovery Using a Restricted Intermediate Language
Publication Type	Conference Paper
Year of Publication	2018
Authors	Pfeffer, T., Herber, P., Druschke, L., Glesner, S.
Conference Name	2018 IEEE 27th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)
ISBN Number	978-1-5386-6916-7
Keywords	automatic analysis, Bianry Analysis, Collaboration, Computational modeling, Conferences, Control Flow Recovery, control-flow recovery, data flow analysis, data-flow analyses, Explosions, formal specification, Human Behavior, human factors, low-level object code, Memory management, Metrics, policy-based governance, program compilers, program diagnostics, program verification, pubcrawl, resilience, Resiliency, restricted control transition intermediate language, Safe Coding, safe control flow recovery, security, security mechanisms, security of data, security policies, Security Validation, Semantics, software fault tolerance, source code level

Abstract

Approaches for the automatic analysis of security policies on source code level cannot trivially be applied to binaries. This is due to the lacking high-level semantics of low-level object code, and the fundamental problem that control-flow recovery from binaries is difficult. We present a novel approach to recover the control-flow of binaries that is both safe and efficient. The key idea of our approach is to use the information contained in security mechanisms to approximate the targets of computed branches. To achieve this, we first define a restricted control transition intermediate language (RCTIL), which restricts the number of possible targets for each branch to a finite number of given targets. Based on this intermediate language, we demonstrate how a safe model of the control flow can be recovered without data-flow analyses. Our evaluation shows that that makes our solution more efficient than existing solutions.

URL	https://ieeexplore.ieee.org/document/8495942
DOI	10.1109/WETICE.2018.00052

Citation Key pfeffer_efficient_2018

automatic analysis Bianry Analysis collaboration Computational modeling Conferences Control Flow Recovery control-flow recovery data flow analysis data-flow analyses Explosions Formal Specification Human behavior Human Factors low-level object code Memory management Metrics policy-based governance program compilers program diagnostics program verification pubcrawl resilience Resiliency restricted control transition intermediate language Safe Coding safe control flow recovery security security mechanisms security of data security policies Security Validation Semantics software fault tolerance source code level