Towards a Certified Lightweight Array Bound Checker for Java Bytecode

David Pichardie
INRIA
Rennes, Bretagne, France
david.pichardie@irisa.fr

Abstract

Dynamic array bound checks are crucial elements for the security of a Java Virtual Machines. These dynamic checks are however expensive and several static analysis techniques have been proposed to eliminate explicit bounds checks. Such analyses require advanced numerical and symbolic manipulations that 1) penalize bytecode loading or dynamic compilation, 2) complexify the trusted computing base. Following the Foundational Proof Carrying Code methodology, our goal is to provide a lightweight bytecode verifier for eliminating array bound checks that is both efficient and trustable. In this work, we define a generic relational program analysis for an imperative, stack-oriented byte code language with procedures, arrays and global variables and instantiate it with a relational abstract domain as polyhedra. The analysis has automatic inference of loop invariants and method pre-/post-conditions, and efficient checking of analysis results by a simple checker. Invariants, which can be large, can be specialized for proving a safety policy using an automatic pruning technique which reduces their size. The result of the analysis can be checked efficiently by annotating the program with parts of the invariant together with certificates of polyhedral inclusions. The resulting checker is sufficiently simple to be entirely certified within the Coq proof assistant for a simple fragment of the Java bytecode language. During the talk, we will also report on our ongoing effort to scale this approach for the full sequential JVM.

E. Denney, T. Jensen (eds.); The 3rd International Workshop on Proof Carrying Code and Software Certification, pp. 29-29