

# Bytecode Verification for Haskell

Robert Dockins and Samuel Z. Guyer

Department of Computer Science  
Tufts University

## Abstract

In this paper we present a method for verifying Yhc bytecode, an intermediate form of Haskell suitable for mobile code applications. We examine the issues involved with verifying Yhc bytecode programs and we present a proof-of-concept bytecode compiler and verifier.

Verification is a static analysis which ensures that a bytecode program is type-safe. The ability to check type-safety is important for mobile code situations where untrusted code may be executed. Type-safety subsumes the critical memory-safety property and excludes important classes of exploitable bugs, such as buffer overflows. Haskell's rich type system also allows programmers to build effective, static security policies and enforce them using the type checker. Verification allows us to be confident the constraints of our security policy are enforced.