Integration Challenges in Static Analysis and Verification

Submitted by Stephen Magill on Fri, 08/14/2020 - 11:59am. Contributor: Stephen Magill

What would it take to reach a point where 80% of developers at DoD Primes and Fortune 500 companies are using formal methods based tools? Certainly tool capabilities, efficiency, and usability play an important role here. But scaling formal methods is not just about scaling analysis techniques to handle more code or more complex specifications, but also about developing the infrastructure to allow analysis tools to deliver value in larger, more complex organizations, and ensuring that analysis processes and workflows can handle systems that evolve over time.

Solving these integration challenges requires addressing technology gaps and supporting organizational and communication patterns that are largely orthogonal to tool-specific scalability concerns. In this talk, we discuss the following question: Can we develop a platform that solves these integration challenges in a way that 1) is tool agnostic (thus benefiting all tools) and 2) supports a variety of program analysis techniques, from bug-finding to sound static analysis and even extending to verification tools?

We will describe the landscape of integration challenges using concrete examples from industry collected from over 30 discussions with developers and multiple tool deployment engagements. We will present affirmative results showing 1) that verification tools can be run in Continuous Integration / Continuous Deployment (CI / CD) workflows and provide continuing value to developers, 2) that certain tool design principles allow a common integration infrastructure to be leveraged by a variety of tools, and 3) that with the right tool support ordinary developers can evolve specifications over time. In addition to these encouraging results, we will describe remaining challenges and highlight open problems (of which there are many). In so doing, we hope to inspire further research in this area and shed light on important aspects of usability that impact adoption but that are largely invisible to formal methods researchers.

This talk covers work on:

- Integration of verification into CI to support both formal correctness proofs and formal methods based regression testing.
- A plugin architecture for formal methods based static analysis tools that supports integration with build systems, code review systems, and developer feedback channels.
- Automated update of information flow specifications to support continuous analysis of evolving codebases.

While the topic of this talk is broader than these specific solutions, these concrete examples will help ground the discussion of what can otherwise be an amorphous set of concerns, and will serve to highlight the open problems that still exist.

Dr. Stephen Magill has spent over 15 years advancing the state of software security and privacy via both basic research and technology transition efforts. Stephen currently serves as CEO of Muse Dev, a company focused on bringing advanced static analysis and verification capabilities to developers. Stephen has led several large research and development projects, including serving as Principal Investigator on a number of DARPA programs, directing research engagements with Amazon Web Services, and leading research for the 2019 State of the Software Supply Chain report. Stephen also serves on the University of Tulsa Industry Advisory Board and numerous program committees and funding panels. He has a Ph.D. in Computer Science from Carnegie Mellon University, and his work has been widely published.
Integration Challenges in Static Analysis and Verification

Presentation HCSS 2020