Edward Pierzchalski

Employment History

Cog Systems

Software engineer, January 2017 to present

- Developed okL4 modules, Linux devices, and Python tools to test for passwords in physical memory on a hypervised Android system, as well as a partitioned, encrypted disk service.
- Migrated a large internal project to use version control appropriately.
- Integrated an embedded Rust project with an okL4 system as part of a well-received internal presentation.

Data61

Research assistant, July 2014 to November 2015, February 2016 to November 2016

- Independently investigating formal semantics of concurrent security-sensitive programs, with a focus on value-dependent classification.
- Developed judgment systems to help automate analysis of security properties in a simple imperative language.
- Verified properties of judgment system using Isabelle.

Google Australia

Engineering intern, November 2015 to February 2016

- Extended collection of public user G+ post data, applied indexing and text salience techniques to extract popular content. Written in Go.
- Demonstrated ability to communicate and coordinate with offsite teams, coding and execution skills, and ability to familiarize myself with internal tools.

NICTA

Software engineering intern, March 2013 to July 2014

- Engineer on the *Perentie* team, an experimental software verification tool written in Scala. Participated in SV COMP 2015, a software verification competition. *Perentie* ranked third in its category after less than a year of development.
- Designed intermediate languages to represent and simplify C program semantics.
- Helped design and implement an embedded domain-specific language for interacting with SMT solvers.

Education

University of New South Wales

- Bachelor of Science (Mathematics), graduated February 2016
- Bachelor of Science (Computer Science), graduated February 2017 with honours

Awards

- Google Third Year Prize
- CSE Undergraduate Performance Prize Year 3, 4th Place

Notable Work

Advanced Algorithmic Verification: Gained experience using Isabelle, an interactive theorem prover. Formally verified a simple C program. Ranked 1st in the course.

Concepts of Programming Languages: Implemented a small Haskell variant, including typechecker. Ranked 1st in the course.

Advanced Operating Systems: Wrote a kernel on top of the seL4 microkernel, written in Rust.

Skills

- Experienced in research and software development, independently or in small to medium teams.
- Strong background on functional programming, language design and semantics, and systems programming.
- Proficient in Rust, C, Haskell, Python, Bash, Isabelle, LaTeX, and Scala.

Publications

Franck Cassez et al. "Perentie: Modular Trace Refinement and Selective Value Tracking". In: *Tools and Algorithms for the Construction and Analysis of Systems*. 2015, pp. 439–442.

Toby Murray et al. "Compositional Verification and Refinement of Concurrent Value-Dependent Noninterference". In: *IEEE Computer Security Foundations Symposium*. Lisbon, Portugal, June 2016.

References

Dr. Toby Murray: Undergraduate thesis supervisor, research lead.
Email: toby.murray@unimelb.edu.au.

Richard Vagg: Principle Engineer at Cog Systems.
Email: richard@cog.systems.