Luke Nelson

luke.r.nels@gmail.com • lukenels.net

Education

University of Washington, Ph.D. Candidate, Computer Science — 2021 – Present

Advisor: Xi Wang

University of Washington, M.S., Computer Science — 2017 – 2021

University of Washington, B.S., Computer Science – *Cum Laude* — 2013 – 2017

Employment

Amazon Web Services, Inc., Applied Scientist — 2022–present

University of Washington, Graduate Research Assistant — 2017–present

University of Washington, Undergraduate Research Assistant — 2016–2017

Facebook, Inc., Software Engineer Intern — Jun. 2016 – Sept. 2016

University of Washington, Undergraduate Research Assistant — 2015–2016

Delphix, Inc., Software Engineer Intern — Jun. 2015 – Sept. 2015

Publications

[1] Sorawee Porncharoenwase, Luke Nelson, Xi Wang, and Emina Torlak. A formal foundation for symbolic evaluation with merging. In *Proceedings of the 49th ACM Symposium on Principles of Programming Languages (POPL)*, Philadelphia, PA, January 2022.

[2] Luke Nelson, Jacob Van Geffen, Emina Torlak, and Xi Wang. Specification and verification in the field: Applying formal methods to BPF just-in-time compilers in the Linux kernel. In *Proceedings of the 14th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pages 41–61, Virtual conference, November 2020.

[3] Luke Nelson, James Bornholt, Arvind Krishnamurthy, Emina Torlak, and Xi Wang. Noninterference specifications for secure systems. *ACM SIGOPS Operating Systems Review*, 54(1):31–39, August 2020.

[4] Jacob Van Geffen, Luke Nelson, Isil Dillig, Xi Wang, and Emina Torlak. Synthesizing JIT compilers for in-kernel DSLs. In *Proceedings of the 32nd International Conference on Computer Aided Verification (CAV)*, pages 564–586, Los Angeles, CA, July 2020.

[5] Luke Nelson, James Bornholt, Ronghui Gu, Andrew Baumann, Emina Torlak, and Xi Wang. Scaling symbolic evaluation for automated verification of systems code with Serval. In *Proceedings of the 27th ACM Symposium on Operating Systems Principles (SOSP)*, pages 225–242, Huntsville, Ontario, Canada, October 2019.

[6] Helgi Sigurbjarnarson, Luke Nelson, Bruno Castro-Karney, James Bornholt, Emina Torlak, and Xi Wang. Nickel: A framework for design and verification of information flow control systems. In *Proceedings of the 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, pages 287–306, Carlsbad, CA, October 2018.

[7] Luke Nelson, Helgi Sigurbjarnarson, Kaiyuan Zhang, Dylan Johnson, James Bornholt, Emina Torlak, and Xi Wang. Hyperkernel: Push-button verification of an OS kernel. In *Proceedings of the 26th ACM Symposium on Operating Systems Principles (SOSP)*, pages 252–269, Shanghai, China, October 2017.

Awards

Best Paper Award and Distinguished Artifact Award, ACM Symposium on Operating Systems Principles — 2019

Corin Anderson Endowed Fellowship, University of Washington — 2017–2018

Presentations and Posters

A proof-carrying approach to building correct and flexible BPF verifiers, *Presentation*
Linux Plumbers Conference 2021 — Sep. 2021

Eliminating bugs in BPF JITs using automated formal verification, *Presentation*
Linux Plumbers Conference 2020 — Aug. 2020

Scaling symbolic evaluation for automated verification of systems code with Serval, *Presentation*
2019 New England Systems Verification Day — Oct. 2019

Verifying enclave systems with Serval, *Presentation*
2019 Open-Source Enclaves Workshop — Jul. 2018

Nickel: A framework for design and verification of information flow control systems, *Presentation*
2018 New England Systems Verification Day — Oct. 2018

Hyperkernel: Push-Button Verification of an OS Kernel, *Poster*
SOSP *(with Helgi Sigurbjarnarson)* — Oct. 2017

Designing Systems for Push-Button Verification, *Presentation*
Allen School 2017 Annual Research Day — Nov. 2017
2017 New England Systems Verification Day *(with Xi Wang and Helgi Sigurbjarnarson)* — Oct. 2017

Ouroboros: Bootstrapping a Formally Verified In-Kernel Interpreter, *Poster*
OSDI (*Jared Roesch, Luke Nelson, Zachary Tatlock, Xi Wang)* — Oct. 2016

Teaching

CSE P 551: Professional Master's Operating Systems — *Teaching Assistant* — Autumn 2019
CSE 551: Graduate Operating Systems — *Teaching Assistant* — Winter 2019
CSE 481A: OS Capstone – *Teaching Assistant* — Winter 2018
CSE 351: The Hardware/Software Interface – *Teaching Assistant* — Spring 2015