# ADAM CHLIPALA

*Cambridge, MA*
*USA*
*adamc@csail.mit.edu*
*http://adam.chlipala.net/*
*A hyperlinked HTML version of this CV is available at http://adam.chlipala.net/cv.html.*

## Research interests

- Design and implementation of computer systems supporting programmer productivity, correctness, security, and performance (including compilers, computer architecture, cryptography, databases, and operating systems)
- Design, implementation, and applications of interactive proof assistants
- Programming languages (broadly construed to include all development tools) and formal methods (for software and hardware)

## Education

- **University of California, Berkeley**
  Electrical Engineering and Computer Science Department
  Computer Science Division
  **Doctor of Philosophy (PhD) in Computer Science**
  8/2003 – 9/2007
  Advisor: George Necula
  Cumulative GPA: 4.0 out of 4.0
  Thesis: Implementing Certified Programming Language Tools in Dependent Type Theory
- **University of California, Berkeley**
  Electrical Engineering and Computer Science Department
  Computer Science Division
  **Master of Science (MS) in Computer Science**
  12/2004
  Advisor: George Necula
  Thesis: An Untrusted Verifier for Typed Assembly Language
- **Carnegie Mellon University**, Pittsburgh, PA
  **Bachelor of Science (BS) in Computer Science with a minor in Mathematical Sciences and University Honors**
  8/2000 – 5/2003
  Cumulative GPA: 4.0 out of 4.0
- **Emmaus High School**, Emmaus, PA
  **High school diploma**
  9/1996 – 6/2000

## Employment

- **Arthur J. Conner (1888) Professor of Computer Science, 7/2024 – ??**
  Professor of Computer Science, 7/2022 – ??
  Associate Professor of Computer Science, 7/2018 – 6/2022
  Associate Professor without Tenure of Computer Science, 7/2015 – 6/2018

Assistant Professor of Computer Science, 7/2011 – 6/2015
Douglas T. Ross (1954) Career Development Professor of Software Technology, 7/2012 – 6/2015
Computer Science and Artificial Intelligence Laboratory
Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology

- **Postdoctoral Fellow, 6/2008 – 6/2011**
  School of Engineering and Applied Sciences
  Harvard University, Cambridge, MA
  Advisor: Greg Morrisett
- **Instructor, 9/2008 – 1/2009**
  COMPSCI 252: Certified Programming with Dependent Types
  School of Engineering and Applied Sciences
  Harvard University, Cambridge, MA
- **OCaml Hacker, 9/2007 – 4/2008**
  Jane Street Capital
- **Graduate Student Researcher, 9/2003 – 8/2007**
  The Open Verifier project
  Computer Science Division
  University of California, Berkeley
  PI: George Necula
- **Instructor, 8/2006 – 12/2006**
  CS294-9: Interactive Computer Theorem Proving
  Computer Science Division
  University of California, Berkeley
- **Research Intern, 6/2005 – 8/2005**
  The Singularity project
  Software Productivity Tools group, Redmond, WA
  Microsoft Research
  Mentor: Manuel Fahndrich
- **Graduate Student Instructor, 1/2005 – 5/2005**
  CS172: Computability and Complexity
  Computer Science Division
  University of California, Berkeley
  Instructor: Brian Lucena
- **Graduate Student Researcher, 6/2003 – 8/2003**
  The BLAST project
  Computer Science Division
  University of California, Berkeley
  PI: Thomas Henzinger
- **Research Assistant, 6/2002 – 5/2003**
  The TILT type-directed Standard ML compiler project
  Computer Science Department
  Carnegie Mellon University, Pittsburgh, PA
  PIs: Robert Harper, Karl Crary

- **Teaching Assistant, 1/2002 – 5/2002**
  15-212: Principles of Programming (introduction to formal reasoning about programs and functional programming with Standard ML)
  Computer Science Department
  Carnegie Mellon University, Pittsburgh, PA
  Instructors: Michael Erdmann, Jeannette Wing
- **Intern/Software Developer, 6/2001 – 8/2001**
  Avaya Communication, Holmdel, NJ
- **Software Developer, Summers, 1998 - 2000**
  Trifecta Technologies, Allentown, PA

## Professional service

- National Science Foundation, panelist, 2012, 2013, 2016, 2021, 2024
- IFIP Working Group on Functional Programming (WG 2.8), member
- IFIP Working Group on Language Design (WG 2.16), member
- International Conference on Certified Programs and Proofs (CPP), steering committee, 2015-present
- 52nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'25), program committee
- First NIST Workshop on Formal Methods within Certification Programs (FMCP'24), co-organizer
- The Tenth International Workshop on Coq for Programming Languages (CoqPL'24), program committee
- Dafny 2024 (Dafny'24), program committee
- 44th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'23), program committee
- 28th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS'23), external review committee
- International Conference on Functional Programming (ICFP), steering committee, 2019-2022
- DARPA Information Science and Technology study group (ISAT), member, 2018-2022
- 49th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'22), virtualization chair
- 15th USENIX Symposium on Operating Systems Design and Implementation (OSDI'21), external review committee
- Interactive Theorem Proving - Twelfth International Conference (ITP'21), program committee
- 42nd ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'21), program committee (area chair)
- 48th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'21), program committee

- 25th ACM SIGPLAN International Conference on Functional Programming (ICFP'20), program chair
- 27th ACM Symposium on Operating Systems Principles (SOSP'19), program committee
- 24th ACM SIGPLAN International Conference on Functional Programming (ICFP'19), external review committee
- 31st International Conference on Computer Aided Verification (CAV'19), program committee
- Interactive Theorem Proving - Ninth International Conference (ITP'18), program committee
- ACM SIGPLAN 2018 Conference on Programming Language Design and Implementation (PLDI'18), program committee
- Web Programming, Design, Analysis, and Implementation track of The Web Conference 2018 (WPDAI'18), program committee
- ACM SIGPLAN Conference on Systems, Programming, Languages and Applications: Software for Humanity (SPLASH'17), workshop-selection program committee
- Interactive Theorem Proving - Eighth International Conference (ITP'17), program committee
- 22nd ACM SIGPLAN International Conference on Functional Programming (ICFP'17), program committee
- 30th IEEE Computer Security Foundations Symposium (CSF'17), program committee
- ACM SIGPLAN 2017 Conference on Programming Language Design and Implementation (PLDI'17), external review committee
- 44th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'17), program committee
- New Directions In Software Technology 2016 (NDIST'16), program co-chair
- 21st ACM SIGPLAN International Conference on Functional Programming (ICFP'16), external review committee
- The Eighth Coq Workshop (Coq-8), program committee
- Interactive Theorem Proving - Seventh International Conference (ITP'16), program committee
- 28th International Conference on Computer Aided Verification (CAV'16), program committee
- 37th IEEE Symposium on Security and Privacy (S&P'16), program committee
- The Second International Workshop on Coq for PL (CoqPL'16), program committee
- 5th International Conference on Certified Programs and Proofs (CPP'16), program co-chair
- 2015 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, & Applications (OOPSLA'15), program committee
- 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS'15), program committee
- POPL'15 Student Research Competition (SRC), judge

- 24th European Symposium on Programming (ESOP'15), program committee
- 9th International Workshop on Logical Frameworks and Meta-languages: Theory and Practice (LFMTP'14), program committee
- 2014 USENIX Annual Technical Conference (USENIX ATC'14), program committee
- Programming Languages meets Program Verification Workshop (PLPV'14), program committee
- Functional Programming Concepts in Domain-Specific Languages (FPCDSL'13), program committee
- 8th International Workshop on Logical Frameworks and Meta-languages: Theory and Practice (LFMTP'13), program committee
- 22nd USENIX Security Symposium (USENIX Security'13), program committee
- Interactive Theorem Proving - Fourth International Conference (ITP'13), program committee
- 43th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN'13), DCCS program committee
- ACM SIGPLAN 2013 Conference on Programming Language Design and Implementation (PLDI'13), external review committee
- 16th International Conference on Foundations of Software Science and Computation Structures (FoSSaCS'13), program committee
- Data Driven Functional Programming Workshop 2013 (DDFP'13), program committee
- 15th International Symposium on Practical Aspects of Declarative Languages (PADL'13), program committee
- 2nd International Conference on Certified Programs and Proofs (CPP'12), program committee
- 7th International Workshop on Logical Frameworks and Meta-languages: Theory and Practice (LFMTP'12), program co-chair
- Interactive Theorem Proving - Third International Conference (ITP'12), program committee
- The Fourth Coq Workshop (Coq-4), program chair
- 24th International Conference on Computer Aided Verification (CAV'12), program committee
- IEEE Symposium on Security & Privacy 2012 (S&P'12), poster chair
- 7th ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'12), program committee
- 39th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'12), program committee
- 16th ACM SIGPLAN International Conference on Functional Programming (ICFP'11), program committee
- 6th International Workshop on Systems Software Verification (SSV'11), program committee
- The Third Coq Workshop (Coq-3), program committee

- Workshop on Foundations of Computer Security (FCS'11), program committee
- 5th International Workshop on Systems Software Verification (SSV'10), program committee
- Mathematically Structured Functional Programming 2010 (MSFP'10), program committee
- The Second Coq Workshop (Coq-2), program committee
- Programming Languages meets Program Verification Workshop (PLPV'10), program committee
- 4th International Workshop on Logical Frameworks and Meta-languages: Theory and Practice (LFMTP'09), program committee
- 3rd Informal ACM SIGPLAN Workshop on Mechanizing Metatheory (WMM'08), program committee
- External reviewer for: ICFP'04, LPAR'05, LICS'06, APLAS'06, TLDI'07, RTA'07, POPL'08, VMCAI'08, PLDI'08, ICFP'08, POPL'09, TLDI'09, ESOP'09, TYPES'08, PLDI'09, ICFP'09, POPL'10, FoSSaCS'10, TACAS'10, MFPS'10, PPDP'10, HOR'10, ICFP'10, POPL'11, VMCAI'11, ESOP'11, PLDI'11, RTA'11, GCM'10, VSTTE'12, FoSSaCS'12, Haskell'12, LFCS'13, ICFP'13, PPDP'13, POPL'14, ESOP'14, ICFP'14, POPL'16, POPL'18, ECOOP'18, POPL'19, PLDI'19, PLDI'22, ICFP'23, POPL'24, POPL'26
- Referee for: CACM, ESL, FI, HOSC, IPL, JACM, JAR, JFP, JFR, SCP, TOPLAS
- External PhD thesis reviewer for: Benjamin Delaware (U. of Texas, Austin), Ronghui Gu (Yale), Andreas Lööw (Chalmers), Brandon Moore (U. of Illinois, Urbana-Champaign), Wilmer Ricciotti (U. of Bologna), Cyril Six (Grenoble), Xiang Wu (Yale), Li-yao Xia (U. Pennsylvania)

## Academic honors

- **SOSP Test of Time Award (for FSCQ)**, 2025
- **German IT Security Award (for CryptOpt)**, 2024
- **Annual "Humies" Awards For Human-Competitive Results Produced By Genetic And Evolutionary Computation (Gold Award, for CryptOpt)**, 2023
- **PLDI Distinguished Reviewer Award**, 2023
- **Burgess (1952) & Elizabeth Jamieson Prize for Excellence in Teaching**, 2023
- **ACM Distinguished Member**, 2019
- **Ruth and Joel Spira Award for Excellence in Teaching**, 2019
- **Most Influential ICFP Paper Award (for PHOAS)**, 2018
- **ACM Senior Member**, 2016
- **National Science Foundation CAREER Award**, 2012
- **National Defense Science and Engineering Graduate Fellowship** winner, 2004
- **National Science Foundation Graduate Research Fellowship** winner, 2004
- **California Microelectronics Fellowship** winner, UC Berkeley EECS Department, 8/2003 – 5/2004
- Inducted into **Phi Kappa Phi**

- Inducted into **Phi Beta Kappa**
- Honorable Mention, **National Science Foundation Graduate Research Fellowship** competition, 2003
- **Andrew Carnegie Scholarship** winner, Carnegie Mellon University, Pittsburgh, PA, 8/2000 – 5/2003

## Citizenship

- American citizen

## Summer schools

- **Summer School on Software Security: Theory to Practice**, University of Oregon, 6/2004

## Software

- **Ur/Web** (http://www.impredicative.com/ur/), a domain-specific programming language design and implementation supporting metaprogramming of web applications with strong static guarantees
- **Cooperative Internet hosting tools** (http://hcoop.sourceforge.net/), including **DomTool** (http://wiki.hcoop.net/DomTool), a domain-specific language in support of shared UNIX system configuration by mutually-untrusting users
- **Dynamic web site tools for Standard ML** (http://smlweb.sourceforge.net/), including separately usable libraries for accessing SQL databases

## Other activities

- Founder of **HCoop, Inc.** (http://hcoop.net/), a democratically run Internet hosting cooperative
- Main administrator and organizer, **Teen Programmers Unite** (http://www.tpu.org/), 1997-2001

## Books

- Adam Chlipala. **Certified Programming with Dependent Types**. MIT Press, 2013. Available online under a Creative Commons license.

## Refereed journal articles

- Arthur Charguéraud, Adam Chlipala, Andres Erbsen, Samuel Gruetter. **Omnisemantics: Smooth Handling of Nondeterminism**. ACM Transactions on Programming Languages and Systems (TOPLAS). Volume 45. Article No. 5, 2023. Association for Computing Machinery.
- Thomas Gregoire, Adam Chlipala. **Mostly Automated Formal Verification of Loop Dependencies with Applications to Distributed Stencil Algorithms**. Journal of Automated Reasoning (JAR). https://doi.org/10.1007/s10817-018-9451-y. Springer-Verlag.

- Andrew W. Appel, Lennart Beringer, Adam Chlipala, Benjamin C. Pierce, Zhong Shao, Stephanie Weirich, Steve Zdancewic. **The Science of Deep Specification**. Philosophical Transactions of the Royal Society A (PTA). 2017 375 20160331. Royal Society.
- Tej Chajed, Haogang Chen, Adam Chlipala, Frans Kaashoek, Nickolai Zeldovich, Daniel Ziegler. **Research Highlight: Certifying a File System using Crash Hoare Logic: Correctness in the Presence of Crashes**. Communications of the ACM (CACM). 60(4). 75-84, 2017. Association for Computing Machinery.
- Adam Chlipala. **Research Highlight: Ur/Web: A Simple Model for Programming the Web**. Communications of the ACM (CACM). 59(8). 93-100, 2016. Association for Computing Machinery.
- Adam Chlipala. **An Introduction to Programming and Proving with Dependent Types in Coq**. Journal of Formalized Reasoning (JFR). 3(2). 1-93, 2010.
- Adam Chlipala. **Modular Development of Certified Program Verifiers with a Proof Assistant**. Journal of Functional Programming (JFP). 18(5/6). 599-647, 2008. Cambridge University Press.

## Refereed conference papers

- Anna Zhang, London Bielicke, Eunice Jun, Adam Chlipala. **Causality and Semantic Separation**. Proceedings of the 46th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'26). June 2026. Conditionally accepted.
- Amanda Liu, Gilbert Bernstein, Shoaib Kamil, Adam Chlipala, Jonathan Ragan-Kelley. **A Mechanized Algebra of Verified Data Structures for Optimizing Sparse Tensor Programs**. Proceedings of the 46th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'26). June 2026. Conditionally accepted.
- Andy Tockman, Pratap Singh, Andres Erbsen, Samuel Gruetter, Adam Chlipala. **Foundational Verification of Running-Time Bounds for Interactive Programs**. Proceedings of the 15th International Conference on Certified Programs and Proofs (CPP'26). January 2026.
- Dustin Jamner, Gabriel Kammer, Ritam Nag, Adam Chlipala. **Pyrosome: Verified Compilation for Modular Metatheory**. Proceedings of the 2025 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, & Applications (OOPSLA'25). October 2025.
- Shixin Song, Tingzhen Dong, Kosi Nwabueze, Julian Zanders, Andres Erbsen, Adam Chlipala, Mengjia Yan. **Securing Cryptographic Software via Typed Assembly Language and a Hardware Extension**. Proceedings of the 2025 ACM SIGSAC Conference on Computer and Communications Security (CCS'25). October 2025.
- Owen Conoly, Andres Erbsen, Adam Chlipala. **Smooth, Integrated Proofs of Cryptographic Constant Time for Nondeterministic Programs and Compilers**. Proceedings of the 46th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'25). June 2025.

- Thomas Bourgeat, Jiazheng Liu, Adam Chlipala, Arvind. **Making Concurrent Hardware Verification Sequential**. Proceedings of the 46th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'25). June 2025.
- Stella Lau, Thomas Bourgeat, Clément Pit-Claudel, Adam Chlipala. **Specification and Verification of Strong Timing Isolation of Hardware Enclaves**. Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS'24). October 2024. *Distinguished Artifact Award*.
- Samuel Gruetter, Thomas Bourgeat, Adam Chlipala. **Verifying Software Emulation of an Unsupported Hardware Instruction**. Proceedings of the Interactive Theorem Proving - Fifteenth International Conference (ITP'24). September 2024.
- Atalay Ileri, Nickolai Zeldovich, Adam Chlipala, Frans Kaashoek. **Probability from Possibility: Probabilistic Confidentiality for Storage Systems Under Nondeterminism**. Proceedings of the 37th IEEE Computer Security Foundations Symposium (CSF'24). July 2024.
- Andres Erbsen, Jade Philipoom, Dustin Jamner, Ashley Lin, Samuel Gruetter, Clément Pit-Claudel, Adam Chlipala. **Foundational Integration Verification of a Cryptographic Server**. Proceedings of the 45th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'24). June 2024.
- Amanda Liu, Gilbert Bernstein, Adam Chlipala, Jonathan Ragan-Kelley. **A Verified Compiler for a Functional Tensor Language**. Proceedings of the 45th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'24). June 2024.
- Samuel Gruetter, Viktor Fukala, Adam Chlipala. **Live Verification in an Interactive Proof Assistant**. Proceedings of the 45th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'24). June 2024.
- Thomas Bourgeat, Ian Clester, Andres Erbsen, Samuel Gruetter, Pratap Singh, Andrew Wright, Adam Chlipala. **Flexible Instruction-Set Semantics via Abstract Monads (Experience Report)**. Proceedings of the 28th ACM SIGPLAN International Conference on Functional Programming (ICFP'23). September 2023.
- Joel Kuepper, Andres Erbsen, Jason Gross, Owen Conoly, Chuyue Sun, Samuel Tian, David Wu, Adam Chlipala, Chitchanok Chuengsatiansup, Daniel Genkin, Markus Wagner, Yuval Yarom. **CryptOpt: Verified Compilation with Random Program Search for Cryptographic Primitives**. Proceedings of the 44th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'23). June 2023. *Distinguished Paper Award*.
- Mohsen Lesani, Li-yao Xia, Anders Kaseorg, Christian J. Bell, Adam Chlipala, Benjamin C. Pierce, Steve Zdancewic. **C4: Verified Transactional Objects**. Proceedings of the 2022 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, & Applications (OOPSLA'22). November 2022.
- Joonwon Choi, Adam Chlipala, Arvind. **Hemiola: A DSL and Verification Tools to Guide Design and Proof of Hierarchical Cache-Coherence Protocols**. Proceedings of the 34th International Conference on Computer-Aided Verification (CAV'22). August 2022.

- Timothy Braje, Alice Lee, Andrew Wagner, Benjamin Kaiser, Daniel Park, Martine Kalke, Robert Cunningham, Adam Chlipala. **Adversary Safety by Construction in a Language of Cryptographic Protocols**. Proceedings of the 35th IEEE Computer Security Foundations Symposium (CSF'22). August 2022. *Lincoln Laboratory Best Paper Award*.
- Jason Gross, Andres Erbsen, Miraya Poddar-Agrawal, Jade Philipoom, Adam Chlipala. **Accelerating Verified-Compiler Development with a Verified Rewriting Engine**. Proceedings of the Interactive Theorem Proving - Thirteenth International Conference (ITP'22). August 2022.
- Jason Gross, Théo Zimmermann, Miraya Poddar-Agrawal, Adam Chlipala. **Automatic Test-Case Reduction in Proof Assistants: A Case Study in Coq**. Proceedings of the Interactive Theorem Proving - Thirteenth International Conference (ITP'22). August 2022.
- Clément Pit-Claudel, Jade Philipoom, Dustin Jamner, Andres Erbsen, Adam Chlipala. **Relational Compilation for Performance-Critical Applications**. Proceedings of the 43rd ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'22). June 2022.
- Amanda Liu, Gilbert Bernstein, Adam Chlipala, Jonathan Ragan-Kelley. **Verified Tensor-Program Optimization Via High-level Scheduling Rewrites**. Proceedings of the 49th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'22). January 2022.
- Mirai Ikebuchi, Andres Erbsen, Adam Chlipala. **Certifying Derivation of State Machines from Coroutines**. Proceedings of the 49th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'22). January 2022.
- Adam Chlipala. **Skipping the Binder Bureaucracy with Mixed Embeddings in a Semantics Course (Functional Pearl)**. Proceedings of the 26th ACM SIGPLAN International Conference on Functional Programming (ICFP'21). August 2021.
- Andres Erbsen, Samuel Gruetter, Joonwon Choi, Clark Wood, Adam Chlipala. **Integration Verification Across Software and Hardware for a Simple Embedded System**. Proceedings of the 42nd ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'21). June 2021.
- Clément Pit-Claudel, Thomas Bourgeat, Stella Lau, Arvind, Adam Chlipala. **Effective Simulation and Debugging for a High-Level Hardware Language Using Software Compilers**. Proceedings of the 26th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS'21). April 2021.
- Clément Pit-Claudel, Peng Wang, Benjamin Delaware, Jason Gross, Adam Chlipala. **Extensible Extraction of Efficient Imperative Programs with Foreign Functions, Manually Managed Memory, and Proofs**. Proceedings of the 9th International Joint Conference on Automated Reasoning (IJCAR'20). June 2020.
- Thomas Bourgeat, Clément Pit-Claudel, Adam Chlipala, Arvind. **The Essence of Bluespec: A Core Language for Rule-Based Hardware Design**. Proceedings of the ACM SIGPLAN 2020 Conference on Programming Language Design and Implementation (PLDI'20). June 2020.

- Benjamin Delaware, Sorawit Suriyakarn, Clément Pit-Claudel, Qianchuan Ye, Adam Chlipala. **Narcissus: Correct-By-Construction Derivation of Decoders and Encoders from Binary Formats**. Proceedings of the 24th ACM SIGPLAN International Conference on Functional Programming (ICFP'19). August 2019.
- Andres Erbsen, Jade Philipoom, Jason Gross, Robert Sloan, Adam Chlipala. **Simple High-Level Code For Cryptographic Arithmetic – With Proofs, Without Compromises**. Proceedings of the IEEE Symposium on Security & Privacy 2019 (S&P'19). May 2019.
- Adam Chlipala. **Algorithmic Checking of Security Arguments for Microprocessors**. Proceedings of the Annual GOMACTech Conference 2019 (GOMACTech'19). March 2019.
- Atalay Ileri, Tej Chajed, Adam Chlipala, Frans Kaashoek, Nickolai Zeldovich. **Proving confidentiality in a file system using DiskSec**. Proceedings of the 13th USENIX Symposium on Operating Systems Design and Implementation (OSDI'18). October 2018.
- Antonis Stampoulis, Adam Chlipala. **Prototyping a Functional Language using Higher-Order Logic Programming: A Functional Pearl on Learning the Ways of Lambda-Prolog/Makam**. Proceedings of the 23rd ACM SIGPLAN International Conference on Functional Programming (ICFP'18). September 2018.
- Benjamin Sherman, Luke Sciarappa, Adam Chlipala, Michael Carbin. **Computable decision-making on the reals and other spaces via partiality and nondeterminism**. Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS'18). July 2018.
- Jason Gross, Andres Erbsen, Adam Chlipala. **Reification by Parametricity: Fast Setup for Proof by Reflection, in Two Lines of Ltac**. Proceedings of the Interactive Theorem Proving - Ninth International Conference (ITP'18). July 2018.
- Haogang Chen, Tej Chajed, Alex Konradi, Stephanie Wang, Atalay Ileri, Adam Chlipala, Frans Kaashoek, Nickolai Zeldovich. **Verifying a High-Performance Crash-Safe File System Using a Tree Specification**. Proceedings of the 26th ACM Symposium on Operating Systems Principles (SOSP'17). October 2017.
- Peng Wang, Di Wang, Adam Chlipala. **TiML: A Functional Language for Practical Complexity Analysis with Invariants**. Proceedings of the 2017 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, & Applications (OOPSLA'17). October 2017.
- Joonwon Choi, Muralidaran Vijayaraghavan, Benjamin Sherman, Adam Chlipala, Arvind. **Kami: A Platform for High-Level Parametric Hardware Specification and its Modular Verification**. Proceedings of the 22nd ACM SIGPLAN International Conference on Functional Programming (ICFP'17). September 2017.
- Adam Chlipala, Benjamin Delaware, Samuel Duchovni, Jason Gross, Clément Pit-Claudel, Sorawit Suriyakarn, Peng Wang, Katherine Ye. **The End of History? Using a Proof Assistant to Replace Language Design with Library Design**. Proceedings of the The 2nd Summit oN Advances in Programming Languages (SNAPL'17). May 2017.
- Ziv Scully, Adam Chlipala. **A Program Optimization for Automatic Database Result Caching**. Proceedings of the 44th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'17). January 2017.

- Thomas Gregoire, Adam Chlipala. **Mostly Automated Formal Verification of Loop Dependencies with Applications to Distributed Stencil Algorithms**. Proceedings of the Interactive Theorem Proving - Seventh International Conference (ITP'16). August 2016.

- Mohsen Lesani, Christian J. Bell, Adam Chlipala. **Chapar: Certified Causally Consistent Distributed Key-Value Stores**. Proceedings of the 43rd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'16). January 2016.

- Haogang Chen, Daniel Ziegler, Tej Chajed, Adam Chlipala, Frans Kaashoek, Nickolai Zeldovich. **Using Crash Hoare Logic for Certifying the FSCQ File System**. Proceedings of the 25th ACM Symposium on Operating Systems Principles (SOSP'15). October 2015. *Best Paper Award and (ten years later) Test of Time Award*.

- Adam Chlipala. **An Optimizing Compiler for a Purely Functional Web-Application Language**. Proceedings of the 20th ACM SIGPLAN International Conference on Functional Programming (ICFP'15). August 2015.

- Muralidaran Vijayaraghavan, Adam Chlipala, Arvind, Nirav Dave. **Modular Deductive Verification of Multiprocessor Hardware Designs**. Proceedings of the 27th International Conference on Computer Aided Verification (CAV'15). July 2015.

- Benjamin Delaware, Clément Pit-Claudel, Jason Gross, Adam Chlipala. **Fiat: Deductive Synthesis of Abstract Data Types in a Proof Assistant**. Proceedings of the 42nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'15). January 2015.

- Adam Chlipala. **Ur/Web: A Simple Model for Programming the Web**. Proceedings of the 42nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'15). January 2015.

- Adam Chlipala. **From Network Interface to Multithreaded Web Applications: A Case Study in Modular Program Verification**. Proceedings of the 42nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'15). January 2015.

- Peng Wang, Santiago Cuellar, Adam Chlipala. **Compiler Verification Meets Cross-Language Linking via Data Abstraction**. Proceedings of the 2014 ACM SIGPLAN International Conference on Object-Oriented Programming, Systems, Languages, & Applications (OOPSLA'14). October 2014.

- Xi Wang, David Lazar, Nickolai Zeldovich, Adam Chlipala, Zachary Tatlock. **Jitk: A Trustworthy In-Kernel Interpreter Infrastructure**. Proceedings of the 11th USENIX Symposium on Operating System Design and Implementation (OSDI'14). October 2014.

- Gregory Malecha, Adam Chlipala, Thomas Braibant. **Compositional Computational Reflection**. Proceedings of the 5th International Conference on Interactive Theorem Proving (ITP'14). July 2014.

- Jason Gross, Adam Chlipala, David Spivak. **Experience Implementing a Performant Category-Theory Library in Coq**. Proceedings of the 5th International Conference on Interactive Theorem Proving (ITP'14). July 2014.

- Adam Chlipala. **The Bedrock Structured Programming System: Combining Generative Metaprogramming and Hoare Logic in an Extensible Program Verifier**. Proceedings of the 18th ACM SIGPLAN International Conference on Functional Programming (ICFP'13). September 2013.
- Thomas Braibant, Adam Chlipala. **Formal Verification of Hardware Synthesis**. Proceedings of the 25th International Conference on Computer Aided Verification (CAV'13). July 2013.
- Adam Chlipala. **Mostly-Automated Verification of Low-Level Programs in Computational Separation Logic**. Proceedings of the ACM SIGPLAN 2011 Conference on Programming Language Design and Implementation (PLDI'11). June 2011.
- Adam Chlipala. **Static Checking of Dynamically-Varying Security Policies in Database-Backed Applications**. Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI'10). October 2010.
- Adam Chlipala. **Ur: Statically-Typed Metaprogramming with Type-Level Record Computation**. Proceedings of the ACM SIGPLAN 2010 Conference on Programming Language Design and Implementation (PLDI'10). June 2010.
- Adam Chlipala. **A Verified Compiler for an Impure Functional Language**. Proceedings of the 37th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'10). January 2010.
- Adam Chlipala, Gregory Malecha, Greg Morrisett, Avraham Shinnar, Ryan Wisnesky. **Effective Interactive Proofs for Higher-Order Imperative Programs**. Proceedings of the 14th ACM SIGPLAN International Conference on Functional Programming (ICFP'09). August 2009.
- Adam Chlipala. **Parametric Higher-Order Abstract Syntax for Mechanized Semantics**. Proceedings of the 13th ACM SIGPLAN International Conference on Functional Programming (ICFP'08). September 2008. *Awarded Most Influential Paper ten years later*.
- Adam Chlipala. **A Certified Type-Preserving Compiler from Lambda Calculus to Assembly Language**. Proceedings of the ACM SIGPLAN 2007 Conference on Programming Language Design and Implementation (PLDI'07). June 2007.
- Adam Chlipala. **Modular Development of Certified Program Verifiers with a Proof Assistant**. Proceedings of the 11th ACM SIGPLAN International Conference on Functional Programming (ICFP'06). September 2006.
- Bor-Yuh Evan Chang, Adam Chlipala, George C. Necula. **A Framework for Certified Program Analysis and Its Applications to Mobile-Code Safety**. Proceedings of the 7th International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI'06). January 2006.
- Dirk Beyer, Adam Chlipala, Thomas Henzinger, Ranjit Jhala, Rupak Majumdar. **Generating Tests from Counterexamples**. Proceedings of the 26th International Conference on Software Engineering (ICSE'04), IEEE Computer Society Press. May 2004.

## Refereed workshop papers

- Haogang Chen, Daniel Ziegler, Adam Chlipala, Frans Kaashoek, Eddie Kohler, Nickolai Zeldovich. **Towards Certified Storage Systems**. Proceedings of the 15th Workshop on Hot Topics in Operating Systems (HotOS'15). May 2015.
- Adam Chlipala. **Position Paper: Thoughts on Programming with Proof Assistants**. Proceedings of the Programming Languages meets Program Verification Workshop (PLPV'06). August 2006.
- Adam Chlipala, George C. Necula. **Cooperative Integration of an Interactive Proof Assistant and an Automated Prover**. Proceedings of the 6th International Workshop on Strategies in Automated Deduction (STRATEGIES'06). August 2006.
- Bor-Yuh Evan Chang, Adam Chlipala, George C. Necula, Robert R. Schneck. **The Open Verifier Framework for Foundational Verifiers**. Proceedings of the 2nd ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'05). January 2005.
- Bor-Yuh Evan Chang, Adam Chlipala, George C. Necula, Robert R. Schneck. **Type-Based Verification of Assembly Language for Compiler Debugging**. Proceedings of the 2nd ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'05). January 2005.
- Adam Chlipala, Leaf Petersen, Robert Harper. **Strict Bidirectional Type Checking**. Proceedings of the 2nd ACM SIGPLAN Workshop on Types in Language Design and Implementation (TLDI'05). January 2005.

## Refereed poster sessions

- Adam Chlipala. **Developing Certified Program Verifiers with a Proof Assistant**. Proceedings of the International Workshop on Proof-Carrying Code (PCC'06). August 2006.

## Invited conference papers

- Dirk Beyer, Adam Chlipala, Thomas Henzinger, Ranjit Jhala, Rupak Majumdar. **The Blast Query Language for Software Verification**. Proceedings of the 11th Static Analysis Symposium (SAS'04), Lecture Notes in Computer Science 3148, Springer-Verlag. August 2004.

## Technical reports

- Adam Chlipala. **Generic Programming and Proving for Programming Language Metatheory**. Technical Report UCB/EECS-2007-147. 2007.
- Adam Chlipala. **Implementing Certified Programming Language Tools in Dependent Type Theory**. Technical Report UCB/EECS-2007-113. 2007.
- Adam Chlipala. **Scrap Your Web Application Boilerplate, or Metaprogramming with Row Types**. Technical Report UCB/EECS-2006-120. 2006.
- Bor-Yuh Evan Chang, Adam Chlipala, George C. Necula. **A Framework for Certified Program Analysis and Its Applications to Mobile-Code Safety**. Technical Report UCB/ERL M05/32. UC Berkeley EECS Department. 2005.

- Adam Chlipala. **An Untrusted Verifier for Typed Assembly Language**. MS Project Report. Technical Report UCB/ERL M04/41. UC Berkeley EECS Department. 2004.

## Talks

- “Scaling Formal Verification to Complete Hardware-Software Stacks”. University at Buffalo CSE Distinguished Speaker Series. September 2025.
- “Research on Different Time Horizons (and an Example with Machine-Checked Proofs)” (keynote talk). PLMW@PLDI’25. June 2025.
- “Programming with Proof Assistants” (invited webinar talk). AIMR’24. April 2024.
- “Fast Cryptographic Code via Partial Evaluation” (invited talk). PEPM’23. January 2023.
- “Fiat Cryptography: A Formally Verified Compiler for Finite-Field Arithmetic”. VSO’22. July 2022.
- “Lectures: From Type Theory to End-to-End Proof of Realistic Systems”. OPLSS’22. June 2022.
- “Fiat Cryptography: A Formally Verified Compiler for Finite-Field Arithmetic”. HCSS’22. May 2022.
- “Correct-by-Construction Cryptography Without Performance Compromises” (invited talk). NUS Computer Science Research Week. January 2022.
- “Fiat Cryptography: A Code-Generation Approach to Correct-by-Construction Cryptography” (invited talk). Verified software: from theory to practice (a workshop of the Isaac Newton Institute). May 2021.
- “Correct-by-Construction Cryptographic Arithmetic in Coq” (invited talk). Lambda Days’21. February 2021.
- “Performance-Scaling Challenges in Formal Verification with Proof Assistants” (invited talk). FLOPS’20. September 2020.
- “Proof Assistants at the Hardware-Software Interface” (invited talk). CPP’20. January 2020.
- “Strong Formal Verification Across a Hardware-Software Stack with RISC-V” (invited talk). SIGARCH Visioning Workshop 2019. June 2019.
- “Fast, Verified Partial Evaluation”. DeepSpec’19. June 2019.
- “Challenges Scaling Type-Theory-Based Verification to Cryptographic Code in Production” (invited talk). TYPES’19. June 2019.
- “Algorithmic Checking of Security Arguments for Microprocessors”. GOMACTech’19. March 2019.
- “Research with RISC-V at MIT CSAIL: Modular Design for Performance, Security, and Formal Verification”. SiFive Technical Symposium, Boston. February 2019.
- “Never Again: Spectre-Proofing Chip Designs with End-to-End Formal Methods”. RISC-V Summit. December 2018.
- “Mechanized Proofs of System Correctness in Production: Cryptography and Beyond” (invited talk). SPLASH-I’18. November 2018.
- “Parametric Higher-Order Abstract Syntax in Fiat Cryptography”. Festschrift for Robert Harper. September 2018.

- “Interactive Proof and the Fall and Rise of the Standard Model of Functional-Program Execution” (invited talk). IFL’18. September 2018.
- “Opportunities to Simplify Computer Systems via Formal Methods”. NEPLS 32. August 2018.
- “How Formal-Methods Adoption Should Drive Changes to System Designs”. Microsoft Research Faculty Summit 2018. August 2018.
- “Lectures: Implementing, specifying, verifying, and compiling hardware components with Kami”. DSSS’18. July 2018.
- “Verification Around the Hardware-Software Interface: Instruction Set, Processors, and Side Channels”. DeepSpec’18. June 2018.
- “Raising the Level of Abstraction in Systems Programming with Fiat and Extensible, Correct-by-Construction Compilers” (invited talk). ENTROPY 2018. January 2018.
- “Kami: Modular Verification of Digital Hardware in Coq”. Inria Paris, Gallium team. January 2018.
- “Fiat Cryptography: Automatic Correct-by-Construction Generation of Low-Level Cryptographic Code”. ETH Zurich, Programming Methodology. January 2018.
- “Coming Soon: Machine-Checked Mathematical Proofs in Everyday Software and Hardware Development”. 34th Chaos Communication Congress. December 2017.
- “Fiat Cryptography: Automatic Correct-by-Construction Generation of Low-Level Cryptographic Code”. Microsoft Research Remond, RiSE group. December 2017.
- “Strong Formal Verification for RISC-V: From Instruction-Set Manual to RTL”. 7th RISC-V Workshop. November 2017.
- “Lectures: Program-specific proof automation”. DSSS’17. July 2017.
- “The End of History? Using a Proof Assistant to Replace Language Design with Library Design”. SNAPL’17. May 2017.
- “Fiat: A New Take on Domain-Specific Languages by Programming with Specifications” (invited talk). RDP’17. January 2017.
- “Fiat: A New Perspective on Compiling Domain-Specific Languages in a Proof Assistant” (invited talk). APLAS’16. November 2016.
- “Rapid Development of Web Applications with Typed Metaprogramming in Ur/Web” (invited talk). SPLASH-I’16. November 2016.
- “Bedrock & Fiat: Specifications and Proofs at the Center of a Programming Ecosystem”. Verified Trustworthy Software Systems (specialist meeting). April 2016.
- “The Science of Deep Specification” (panel). Verified Trustworthy Software Systems (public meeting). April 2016.
- “Fiat: Extensible Code Generation with Proofs” (invited talk). PEPM’16. January 2016.
- “Ur/Web: A Simple Model for Programming the Web”. Mozilla San Francisco. January 2016.
- “Lectures: Formal Proof for C-Like Programs”. MITx online course: Cybersecurity: Technology, Application and Policy. September 2015.

- “An Optimizing Compiler for a Purely Functional Web-Application Language”. ICFP’15. August 2015.
- “Phantom Monitors: A Simple Foundation for Modular Proofs of Fine-Grained Concurrent Programs”. IMDEA Software. July 2015.
- “Lectures: The Coq Proof Assistant and Its Applications to Programming-Language Semantics”. OPLSS’15. June 2015.
- “Bedrock: A Clean-Slate Platform for Developing Verified Software Inside a Proof Assistant” (invited talk). CoqPL’15. January 2015.
- “From Network Interface to Multithreaded Web Applications: A Case Study in Modular Program Verification”. POPL’15. January 2015.
- “Ur/Web: A Simple Model for Programming the Web”. POPL’15. January 2015.
- “Proof Engineering: Implementation Challenges in Rigorously Verified Software” (invited talk). PLMW’15. January 2015.
- “Bedrock: A Software Development Ecosystem Inside a Proof Assistant”. Microsoft Research Cambridge, PPT Group. December 2014.
- “Correct-by-Construction Program Synthesis in Coq” (invited talk). TPP’14. December 2014.
- “Ur/Web: A Simple Model for Programming the Web”. Kyoto University RIMS. December 2014.
- “Ur/Web: A Simple Model for Programming the Web”. Boston Haskell. August 2014.
- “Bedrock: A Foundational Proof-Carrying Code Platform with Functional Correctness Proofs” (invited talk). IHP Workshop on Certification of High-Level and Low-Level Programs. July 2014.
- “Ur/Web: Streamlined Web Apps via Fancy Types” (invited talk). Twitter, Inc., San Francisco. January 2014.
- “Ur/Web: Taking Syntax Seriously” (invited talk). SCRIPT’13. November 2013.
- “Adventures in Knot-Tying while Verifying a Thread Library in Coq”. HOPE’13. September 2013.
- “The Bedrock Structured Programming System: Combining Generative Metaprogramming and Hoare Logic in an Extensible Program Verifier”. ICFP’13. September 2013.
- “A Taste of Effective Coq Proof Automation” (invited tutorial). POPL’13 TutorialFest. January 2013.
- “Web Security via Types and Theorem-Proving in the Ur/Web Programming Language”. CSAIL Student Workshop. September 2011.
- “Web Security via Types and Theorem-Proving in the Ur/Web Programming Language”. IBM Watson Research Center. August 2011.
- “Bedrock: Higher-Order and Automated Proofs about Low-Level Programs” (invited talk). LOLA’11. June 2011.
- “Ur/Web, a Domain-Specific Functional Programming Language for Modern Web Applications”. UC Berkeley. June 2011.

- “Mostly-Automated Verification of Low-Level Programs in Computational Separation Logic”. PLDI’11. June 2011.
- “Ur/Web, a Domain-Specific Functional Programming Language for Modern Web Applications”. MIT PL Working Group. December 2010.
- “Static Checking of Dynamically-Varying Security Policies in Database-Backed Applications”. OSDI’10. October 2010.
- “Foundational Program Verification in Coq with Automated Proofs” (invited tutorial). MSFP’10. September 2010.
- “Ur/Web, a Domain-Specific Functional Programming Language for Modern Web Applications”. COPLAS, ITU Copenhagen. August 2010.
- “Ur/Web: A Statically-Typed Language for Building Web Applications from Components” (invited talk). Emerging Languages Camp 2010. July 2010.
- “A Bottom-Up Approach to Safe Low-Level Programming” (invited talk). MLPA’10. July 2010.
- “Generating Pieces of Web Applications with Type-Level Programming”. DTP’10. July 2010.
- “Ur: Statically-Typed Metaprogramming with Type-Level Record Computation”. PLDI’10. June 2010.
- “Safe Database Abstractions with Type-Level Record Computation” (invited talk). RADICAL’10. May 2010.
- “A Sane Approach to Modern Web Application Development”. Boston Lisp. February 2010.
- “A Verified Compiler for an Impure Functional Language”. POPL’10. January 2010.
- “Towards the Ultimate Web Application Framework, via Fancy Types”. New England F# User Group. November 2009.
- “Syntactic Proofs of Compositional Compiler Correctness”. NJPLS. October 2009.
- “Metaprogramming AJAX Apps with Static Types”. DEFUN’09. September 2009.
- “Engineering a Verified Functional Language Compiler” (invited talk). WMM’09. September 2009.
- “Effective Interactive Proofs for Higher-Order Imperative Programs”. ICFP’09. August 2009.
- “Metaprogramming AJAX Apps with Static Types”. Microsoft Research Redmond. July 2009.
- “Liberating Semi-Automated PL Proofs from Binder Bookkeeping”. Northeastern University Programming Languages Seminar. February 2009.
- “Liberating Semi-Automated PL Proofs from Binder Bookkeeping”. Boston University Programming Languages Reading Group. February 2009.
- “Statically-Checked Metaprogramming for Web Applications”. NEPLS 21. November 2008.
- “Parametric Higher-Order Abstract Syntax for Mechanized Semantics”. ICFP’08. September 2008.
- “Generic Programming and Proving for Programming Language Metatheory”. WMM’07. October 2007.

- “A Certified Type-Preserving Compiler from Lambda Calculus to Assembly Language”. PLDI’07. June 2007.
- “A Certified Type-Preserving Compiler from Lambda Calculus to Assembly Language”. Open Source Quality Project Retreat. May 2007.
- “A Certified Type-Preserving Compiler from Lambda Calculus to Assembly Language”. Projet Gallium seminar. January 2007.
- “Modular Development of Certified Program Verifiers with a Proof Assistant”. ICFP’06. September 2006.
- “Position Paper: Thoughts on Programming with Proof Assistants”. PLPV’06. August 2006.
- “Cooperative Integration of an Interactive Proof Assistant and an Automated Prover”. STRATEGIES’06. August 2006.
- “Developing Sound Program Analysis Tools by Programming with Proofs”. Open Source Quality Project Retreat. May 2006.
- “A Framework for Certified Program Analysis and Its Applications to Mobile-Code Safety”. VMCAI’06. January 2006.
- “Proof-Carrying Verifiers”. Open Source Quality Project Retreat. May 2005.
- “The Open Verifier Framework for Foundational Verifiers”. TLDI’05. January 2005.

## Talks by coauthors (without papers)

- Joel Kuepper, Andres Erbsen, Jason Gross, Owen Conoly, Chuyue Sun, Samuel Tian, David Wu, Adam Chlipala, Chitchanok Chuengsatiansup, Daniel Genkin, Markus Wagner, Yuval Yarom. “CryptOpt: Verified Compilation with Random Program Search for Cryptographic Primitives”. Real World Cryptography 2023. March 2023.
- Dustin Jamner, Gabriel Kammer, Adam Chlipala. “Pyrosome: A Framework for Modular, Extensible, Equivalence-Preserving Compilation”. The Ninth International Workshop on Coq for Programming Languages. January 2023.
- Samuel Gruetter, Adam Chlipala, Arthur Charguéraud, Andres Erbsen. “Omnisemantics: Smooth Handling of Nondeterminism”. The Ninth International Workshop on Coq for Programming Languages. January 2023.
- Samuel Gruetter, Adam Chlipala, Arthur Charguéraud, Andres Erbsen. “Omnisemantics: Smooth Handling of Nondeterminism”. 33rd New England Programming Languages and Systems Symposium. November 2022.
- Shardul Chiplunkar, Clément Pit-Claudel, Adam Chlipala. “Automated Synthesis of Verified Firewalls”. The Seventh International Workshop on Coq for Programming Languages. January 2021.
- Samuel Gruetter, Andres Erbsen, Joonwon Choi, Adam Chlipala. “Formal methods for hardware-software integration on RISC-V embedded systems”. 2019 RISC-V Summit. December 2019.
- Andres Erbsen, Jade Philipoom, Jason Gross, Robert Sloan, Adam Chlipala. “Efficient and Verified Finite-Field Operations”. Real World Cryptography 2019. March 2019.

- Benjamin Sherman, Luke Sciarappa, Adam Chlipala, Michael Carbin. “Programming with Continuous Spaces”. 30th New England Programming Languages and Systems Symposium. October 2016.
- Christian J. Bell, Mohsen Lesani, Adam Chlipala. “Phantom Monitors: A Simple Foundation for Modular Proofs of Fine-Grained Concurrent Programs”. 28th New England Programming Languages and Systems Symposium. November 2015.
- Mohsen Lesani, Anders Kaseorg, Christian J. Bell, Adam Chlipala. “Composable Modular Atomicity”. 28th New England Programming Languages and Systems Symposium. November 2015.
- Gregory Malecha, Adam Chlipala, Patrick Hulin, Edward Z. Yang. “A Framework for Verifying Low-level Programs”. 26th New England Programming Languages and Systems Symposium. December 2012.

## Invited participation in workshops

- 8th High-Assurance Crypto Software Workshop. March 2024.
- 7th High-Assurance Crypto Software Workshop. March 2023.
- 6th High-Assurance Crypto Software Workshop. April 2022.
- 5th High-Assurance Crypto Software Workshop. April 2021.
- 4th High-Assurance Crypto Software Workshop. January 2020.
- 1st High Assurance Systems Engineering Workshop. January 2019.
- Microsoft Research Faculty Summit 2018. August 2018.
- 3rd High-Assurance Crypto Software Workshop. January 2018.
- SAP HANA TechDays. August 2017.
- Google Academic Security and Privacy Research Summit. June 2017.
- NII Shonan Seminar #98: Language integrated queries: towards standard logics for big data analytics. May 2017.
- ISAT Workshop: Augmented Developers: Tools for Hybrid Man-Machine Software Engineering. February 2017.
- 2nd High-Assurance Crypto Software Workshop. January 2017.
- 1st High-Assurance Crypto Software Workshop. January 2016.
- 2nd Core Infrastructure Workshop (Linux Foundation). July 2015.
- Dagstuhl Seminar #15191: Compositional Verification Methods for Next-Generation Concurrency. May 2015.
- 1st Core Infrastructure Workshop (Linux Foundation). January 2015.
- Dagstuhl Seminar #10351: Modelling, Controlling and Reasoning About State. August 2010.

## Research funding

- “FMitF: Track I: Formally Verified Programmable Network Switches”, PI, NSF FMitF

- “Collaborative Research: SHF: Medium: High-Performance, Verified Accelerator Programming”, PI, NSF CCF
- “PPoSS: LARGE: Intel: Combining Learning and Formal Verification for Scalable Machine Programming (ScaMP)”, co-PI, NSF PPoSS (with Intel)
- “SaTC: CORE: Small: Scaling Correct-by-Construction Code Generation for Cryptography”, PI, NSF SaTC
- “SaTC: CORE: Medium: Provably Secure, Usable, and Performant Enclaves in Multicore Processors”, co-PI, NSF SaTC
- “Tools for Correct-by-Construction Hardware and Software in Critical Systems”, Sandia National Laboratories
- “ACHILLES: Assured CryptograpHic Integration of muLtiple Languages for Encrypted Systems”, co-PI for sub, IARPA HECTOR program
- “The Hardware Security Compiler: A Rapid-Development Workflow with End-to-End Formal Verification”, PI for prime, DARPA SSITH program
- “FMitF: Verifying Concurrent System Software with Cspec”, co-PI, NSF CCF
- “SaTC: CORE: Small: verifying security for data non-interference”, co-PI, NSF CNS
- “Correct-by-Construction and Automatic Generation of Elliptic Curve Cryptography Primitives”, Google Research Award
- “CSR: Medium: A High-Performance Certified File System and Applications”, co-PI, NSF CNS
- “RINGS: Regenerative, INtent-Guided Systems”, PI for sub, DARPA BRASS program
- “Collaborative Research: Expeditions in Computing: The Science of Deep Specification”, PI for MIT, NSF Expeditions in Computing
- “SHF: Medium: Fiat: Correct-by-Construction and Mostly Automated Derivation of Programs with an Interactive Theorem Prover”, PI, NSF CCF
- “A Trust Anchor Secure Language via the Bedrock Platform”, PI, Google ATAP Trust Anchor program
- “Cybersecurity project”, QCRI-CSAIL joint program
- “CAREER: A Formal Verification Platform Focused on Programmer Productivity”, PI, NSF CCF
- “SHF: Small: Capitalizing on First-Class SQL Support in the Ur/Web Programming Language”, PI, NSF CCF
- “CAP3: A Computer Aided Performance Programming Platform”, co-PI, DoE X-Stack program
- “CARS: A Platform for Scaling Formal Verification to Component-Based Vehicular Software Stacks”, PI for sub, DARPA HACMS program
- “Safe but Unsandboxed Native Code in the Browser”, Google Research Award