

Education

- 2016–2020 **PhD, Computer Science**, *Australian National University*, Canberra, Australia
2004–2009 **Integrated Post Graduate**, *Indian Institute of Information Technology & Management*, Gwalior, India

PhD thesis


- Title *Formally Verified Verifiable Electronic Voting Scheme*
Supervisor Dirk Pattinson
Description We focussed on three main challenges posed by electronic voting: correctness, privacy, and verifiability. We addressed correctness by using a theorem prover to implement the vote counting algorithm, privacy by using homomorphic encryption, and verifiability by generating a independently checkable scrutiny sheet. Our work has been carried out in Coq theorem prover.
- 2021- **Senior Research Fellow**, *University of Cambridge*, Cambridge, United Kingdom
I am working on formalising network protocols framework to ensure their safety and security. The goal is to develop a mathematical framework, proven correct in a theorem prover, so that a protocol designer can assess the properties of their design using my framework
- 2020–21 **Research Fellow**, *University of Melbourne*, Melbourne, Australia
I worked with Toby Murray on *Security Concurrent Separation Logic*. The aim was to formally (mathematically) reason about memory safety and information flow property of concurrent programs, written in C.
- 2013–2015 **Lecturer**, *International Institute of Information Technology*, Bhubaneswar, India
This role was primarily geared towards teaching, and the courses I taught were *C programming* and *Cryptography*. In addition, every year I supervised two master's students in their final year project.
- 2012–2013 **Haskell Developer**, *Parallel Scientific*, Colorado, USA
In this role, my primary job was research and prototype high performance software programs, mainly linear algebra algorithms written in Haskell.
- 2009–2012 **Technical Assistant**, *Government of India*, Kolkata, India
I worked as a contract developer for automating the day-to-day job, including enforcing the security policies of the organisation.
- 2008–2008 **Summer Intern**, *Arcelor Mittal, Research & Development Technological Centre*, Avilés, Spain
During this role, I worked on formalising many business requirements into a linear programming problem and wrote a custom interface that interacted with their in-house linear programming solver.

Skills

Coding Coq, Haskell, OCaml, Lean, Python, C

📞 +44-7824648138

✉ mt883@cam.ac.uk, mukesh.tiwari@anu.edu.au, mukeshtiwari.iiitm@gmail.com

in [mukesh-tiwari-3609486/](https://github.com/mukesh-tiwari) •  [mukeshtiwari](https://github.com/mukesh-tiwari)

Language Hindi, English

Awards

HDR Fee Remission Merit Scholarship

ANU PhD Scholarship (International)

Full Scholarship to attend DeepSpec Summer School 2018, Princeton University


Travel Scholarship to attend Marktoberdorf Summer School 2019

Publications

- [1] Mukesh Tiwari and Dirk Pattinson. Machine Checked Properties of the Schulze Method. In *7th Workshop on Hot Issues in Security Principles and Trust*, 2021.
- [2] Nadim Kobeissi, Georgio Nicolas, and Mukesh Tiwari. Verifpal: Cryptographic Protocol Analysis for the Real World. In Karthikeyan Bhargavan, Elisabeth Oswald, and Manoj Prabhakaran, editors, *Progress in Cryptology – INDOCRYPT 2020*, pages 151–202, Cham, 2020. Springer International Publishing.
- [3] Thomas Haines, Rajeev Goré, and Mukesh Tiwari. Verified Verifiers for Verifying Elections. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, CCS '19*, page 685–702, New York, NY, USA, 2019. Association for Computing Machinery.
- [4] Thomas Haines, Dirk Pattinson, and Mukesh Tiwari. Verifiable Homomorphic Tallying for the Schulze Vote Counting Scheme. In *Verified Software: Theories, Tools, and Experiments*. Springer, 2019.
- [5] Milad K. Ghale, Rajeev Goré, Dirk Pattinson, and Mukesh Tiwari. Modular Formalisation and Verification of STV Algorithms. In Robert Krimmer, Melanie Volkamer, Véronique Cortier, Rajeev Goré, Manik Hapsara, Uwe Serdült, and David Duenas-Cid, editors, *Electronic Voting*, pages 51–66, Cham, 2018. Springer International Publishing.
- [6] Lyria Bennett Moses, Rajeev Goré, Ron Levy, Dirk Pattinson, and Mukesh Tiwari. No More Excuses: Automated Synthesis of Practical and Verifiable Vote-Counting Programs for Complex Voting Schemes. In Robert Krimmer, Melanie Volkamer, Nadja Braun Binder, Norbert Kersting, Olivier Pereira, and Carsten Schürmann, editors, *Electronic Voting*, pages 66–83, Cham, 2017. Springer International Publishing.
- [7] Dirk Pattinson and Mukesh Tiwari. Schulze Voting as Evidence Carrying Computation. In Mauricio Ayala-Rincón and César A. Muñoz, editors, *Interactive Theorem Proving*, pages 410–426, Cham, 2017. Springer International Publishing.
- [8] R. Choudhari, K. V. Arya, M. Tiwari, and K. S. Choudhary. Performance Evaluation of SCTP-Sec: A Secure SCTP Mechanism. In *2009 Fourth International Conference on Computer Sciences and Convergence Information Technology*, pages 1111–1116, Nov 2009.
- [9] M. Tiwari, K. V. Arya, R. Choudhari, and K. S. Choudhary. Designing Intrusion Detection to Detect Black Hole and Selective Forwarding Attack in WSN Based on

☎ +44-7824648138

✉ mt883@cam.ac.uk, mukesh.tiwari@anu.edu.au, mukeshtiwari.iiitm@gmail.com

in [mukesh-tiwari-3609486/](https://www.linkedin.com/in/mukesh-tiwari-3609486/) •  [mukeshtiwari](https://github.com/mukeshtiwari)

Local Information. In *2009 Fourth International Conference on Computer Sciences and Convergence Information Technology*, pages 824–828, Nov 2009.

Work in Progress


- 1 Verified Secure Declassification for Concurrent Applications. In this work, we develop a formal model of leaking sensitive data.
- 2 Formally Verified Verifiable Group Generators. In this work, we develop a formally verified algorithm that can be used to bootstrap a democratic election.
- 3 Modelling Networking Protocols Mathematically. In this work, we develop a formally verified framework that a network protocol designer can use to verify the security properties of their protocol
- 4 Machine Checking the Bayer-Groth Proof of Shuffle. In this work, we formalise the Bayer-Groth Proof of Shuffle, used in many democratic elections to shuffle the ballot.
- 5 Theorem Provers to Protect Democracies. In this work, we are formalising all the cryptographic components, written in Java, used in Swiss Post in Coq theorem prover. Our goal is to write an independent verifier for scrutine sheet of elections conducted by Swiss Post software programs.

References

- Dirk Pattinson, Research School of Computer Science, Australian National University, Canberra, dirk.pattinson@anu.edu.au
- Timothy Griffin, Computer Laboratory, University of Cambridge, Cambridge, tgg22@cam.ac.uk
- Thomas Haines, Research School of Computer Science, Australian National University, Canberra, thomas.haines@anu.edu.au
- Toby Murray, School of Computing and Information Systems, University of Melbourne, Melbourne, toby.murray@unimelb.edu.au
- Véronique Cortier, Research Director, CNRS, LORIA, Nancy, France, veronique.cortier@loria.fr

☎ +44-7824648138

✉ mt883@cam.ac.uk, mukesh.tiwari@anu.edu.au, mukeshtiwari.iiitm@gmail.com

in [mukesh-tiwari-3609486/](#) •  [mukeshtiwari](#)