Research Statement

Mukesh Tiwari

My research interests lie in constructing correct software programs used in a democratic process, e.g., voting. During my PhD, I worked on formal verification of the Schulze method –a preferential vote-counting method– [1, 2] in the Coq theorem prover. In my Coq implementation, I assumed that (preferential) ballots were in plaintext, but preferential ballots admit "Italian" attack [3, 4]. Although my previous two Coq implementations [1, 2] satisfied correctness and verifiability criteria, they lacked privacy due to "Italian" attack. Therefore, to avoid the "Italian" attack on the Schulze method, I used a homomorphic encryption to encrypt the ballots and computed the winner by combining all the (encrypted) ballots, without decrypting any individual ballot (privacy) [5]. The downside of an encrypted ballot Schulze election, or in fact any encrypted ballot election, is difficulty in auditing because of involved mathematics of cryptography. Therefore, in the future I want to explore the possibility of a verified prototype of scrutiny-sheet checker for encrypted ballots Schulze elections. Finally, I worked on a verified prototype of a simple approval voting election scrutiny-sheet checker for International Association of Cryptologic Research (IACR) [6]. In addition, I was involved in the formalisation of single transferable vote, used in the Australian Senate [7].

My long-term aim is to make formal verification ubiquitous in software development, specifically for the software programs deployed in public domain that affect common people. My expertise in **Theorem Proving, Cryptography, and Election Security** gives me an unique perspective to solve challenging problems that matter to many democracies and its citizens.

1 Research Integration Project

Throughout my research career, I have secured softwares by proving them mathematically correct using the Coq theorem prover, and at CentraleSupélec (Rennes campus) I would like to use my formal verification expertise to strengthen the capability of **CIDRE** team in formal methods for security. For example, I can be contribute to the ongoing formal verification projects, e.g., FreeSpec, etc. In addition, I am open to explore other areas of security, e.g., network security, malware analyses, embedded system security, etc.

2 Future Work

In addition to contributing to the existing projects at **CIDRE**, I would like to work on following projects to add more capability at **CIDRE** and forge research collaboration with other research groups in France and worldwide.

2.1 Mathematically Proven Correct Cryptographic Algorithms

Cryptographic algorithms are used ubiquitously to secure the data, and correctness is an utmost requirement for any cryptographic algorithm implementation. Therefore, I will focus on developing mathematically proven correct cryptographic algorithms used in electronic voting, blockchain, and secure communication, e.g., sigma protocols (zero-knowledge-proof), verifiable (shuffling) mix-networks, multi-party computations, secret sharing, zk-snark, etc. The rationale behind implementing these algorithms is that anyone can use them to construct an utility, e.g., an election scrutiny-sheet checker, a vote-tallying system based on blockchain, a verifiable ballot mixing service, an auction server, etc. One of the motivation behind this project is to replace the SwissPost Java implementations[1] with mathematically proven correct Coq implementations[2].

2.2 Mathematically Proven Correct Vote-Counting Algorithms

In future, I will focus on developing mathematically proven correct software programs for vote-counting methods used across the world such as *Single Transferable Vote (STV), First Past the Post (FPTP), Instant-runoff voting(IRV), etc.,* in the Coq theorem prover. All the vote-counting methods, by design, lend themselves well to computing the winner from plaintext ballots; however, so far there is very little research in computing the winner from encrypted ballots, while ensuring correctness, privacy, and verifiability. Therefore, producing the winner from encrypted ballots is a challenging task. The motivation for this project is that once we have mathematically proven correct components, anyone –election commission or members of general public– can use them to conduct elections, referendums, and verify elections' outcome without worrying about software bugs. The cryptographic algorithms formalised in the previous step are going to be used as a building block in this project.

2.3 Mathematically Proven Correct Decentralised Application

I will focus on a mathematically proven correct decentralised peer-to-peer technical solution [8, 9, 10, 11] in the Coq theorem prover. The motivation is to help whistleblowers in leaking documents and exposing corruption without revealing their identities. Being vocal against the government is one the most fundamental right of any citizen, but many authoritative governments do not appreciate dissent of any form. Therefore, it uses its powerful machinery to punish dissidents, in the name of national security. The inspiration for this project comes from David McBridge[3] and Richard Boyle[4]. David McBride is facing a threat of lifetime jail after leaking the material alleging war crimes by members of the Australia's Special Operations Task Group in Afghanistan, while Richard Boyle is facing 161 years for exposing the corruption inside the Australian Taxation office (Australia is ranked very high in democracy index[5]). This research will open the door of collaboration with many groups working in verified networking, and verified distributed systems.

[1] https://bit.ly/3EODmnF
[2] An ongoing project https://github.com/mukeshtiwari/Dlog-zkp/
[3] https://en.wikipedia.org/wiki/David_McBride_(whistleblower)
[4] https://bit.ly/3OQ6kbC
[5] https://worldpopulationreview.com/country-rankings/democracy-countries

2.4 Mathematically Proven Correct Social Choice Properties

Computational social choice theory is a research area that is concerned with aggregation of ballots (preferences) of multiple voters (agents) and encompasses computer science, mathematics, economics, and political science. Typical applications of computational social choice theory is voting (preference aggregation), resource allocation, and fair division. Most of the proofs in computational social choice theory are pen-and-paper proofs, and one of my long term future research goal is to make them more precise using the Coq theorem prover [12]. Moreover, I would also focus on designing voting algorithms. This research opens the door of collaboration with political scientists, social choice theorists, economists, and game theorists.

References

[1] 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.

[2] 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 *International Joint Conference on Electronic Voting*, pages 66–83. Springer, 2017.

[3] J. Otten. Fuller disclosure than intended. 2003. Accessed on October 17, 2019.

[4] Josh Benaloh, Tal Moran, Lee Naish, Kim Ramchen, and Vanessa Teague. Shuffle-sum: coercion-resistant verifiable tallying for STV voting. *IEEE Trans. Information Forensics and Security*, 4(4):685–698, 2009.

[5] Thomas Haines, Dirk Pattinson, and Mukesh Tiwari. Verifiable Homomorphic Tallying for the Schulze Vote Counting Scheme. In Supratik Chakraborty and Jorge A. Navas, editors, *Verified Software. Theories, Tools, and Experiments*, pages 36–53, Cham, 2020. Springer International Publishing.

[6] 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.

[7] 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.

[8] Joseph K Liu, Victor K Wei, and Duncan S Wong. Linkable Spontaneous Anonymous Group Signature for Ad Hoc Groups. In *Australasian Conference on Information Security and Privacy*, pages 325–335. Springer, 2004.

[9] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W. Hong. *Freenet: A Distributed Anonymous Information Storage and Retrieval System*, pages 46–66. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.

[10] zzz (Pseudonym) and Lars Schimmer. Peer Profiling and Selection in the I2P Anonymous Network. In *PetCon 2009.1*, pages 59–70, 2009.

[11] Henry Corrigan-Gibbs and Bryan Ford. Dissent: Accountable anonymous group messaging. In *Proceedings of the 17th ACM Conference on Computer and Communications Security*, CCS '10, page 340–350, New York, NY, USA, 2010. Association for Computing Machinery.

[12] Mukesh Tiwari and Dirk Pattinson. Machine Checked Properties of the Schulze Method. In *7th Workshop on Hot Issues in Security Principles and Trust*, 2021.