Protecting privacy at the ballot box with secure multiparty computation

June 6 2019, by Brian Huchel

Computer science professor Tiark Rompf, left, and principal investigator Milind Kulkarni, an electrical and computer engineering professor, are part of a project that will combine programming languages and security research to help build computational trust. Their project, called HACCLE, is receiving fuding from the Intelligence Advanced Research Activity, an organization within the Office of the Director of National Intelligence. Credit: Purdue University photo/Vincent Walter

Shortly after the start of the new year, Americans around the nation will start returning to polling stations to vote in presidential primaries. How confident they feel in the voting process could depend on something called "secure multiparty computation."

Secure multiparty computation allows different sides to work together and solve a problem. It can, for instance, keep bids private during an auction or guarantee privacy during election voting. Most importantly, it allows for trust on social and commercial interactions.

Computer engineering and computer science researchers at Purdue are working together to find the answer.

The project, named High Assurance Compositional Cryptography: Languages and Environments (HACCLE), is working to utilize the areas of security and programming language to solve the questions and hurdles created by current methods.

"The challenge is that there are a wide range of questions that need to be addressed when developing those applications," said Milind Kulkarni, an associate professor in electrical and computer engineering, who is leading the project. "And every time you answer these questions differently, it takes a heroic effort from cryptographic experts to deliver an effective solution."

Kulkarni said the Purdue researchers are working to take the task of developing secure multiparty applications out of the realm of experts and make it accessible to ordinary programmers.

HACCLE is intended to provide programming languages and the verification, optimization, and execution tools to address the challenges currently encountered. The ultimate goal is to allow programmers to write secure multiparty computation applications with minimum effort and maximum performance.

Additional faculty involved in project research are Tiark Rompf, Roopsha Samanta, Hemanta Maji, Aniket Kate, Christina Garman, Benjamin Delaware and Jeremiah Blocki, all professors in Purdue's Department of Computer Science. The group is collaborating with Reservoir Labs, a technology and solutions company in New York City.

The project recently was awarded grant funding by the Intelligence Advanced Research Projects Activity, an organization within the Office of the Director of National Intelligence. Through IARPA's Homomorphic Encryption Computing Techniques with Overhead Reduction, the HACCLE project will receive up to $10.7 million.

Currently, a number of issues can come up when two or more parties attempt secure computation. These range from how much do the parties need to communicate to what specific cryptographic techniques will work best to implement this work.

"That's where the programming languages magic comes in," Kulkarni said. "Faculty in the programming languages and security areas, both strengths at Purdue, are designing new domain-specific languages to write secure multiparty computation applications."

Rompf says the technology will cross disciplines like never before.

"This project is especially exciting because it is building bridges in multiple ways," he said. "First, between programming languages and security research and the respective communities. Second, on campus between the College of Science and the College of Engineering. True progress can only be made by crossing boundaries, and this is why we will be successful."

Blocki, an assistant professor of computer science, says the new tools can allow organizations or individuals to cooperate in fundamentally new ways, even if they distrust one another.

"Secure multiparty computation allows our (mutually distrusting) parties to compute any function of their joint data without exposing any confidential data," he says.

He explains it this way: Say there is a group of people, and the goal is to figure out if there are any romantic pairings. Each person writes in the name of their crush. If two share a crush, both parties will be notified at the end of the protocol. If not, no one will learn of what a person entered.

"As part of the project we are working to improve the core cryptographic primitives that are used in secure multiparty computation. We also want to develop efficient techniques to help compose these primitives in a provably secure way and to estimate the overhead of the final protocol when we combine several different cryptographic primitives," Blocki said. "Ultimately, the goal is to develop a compiler that allows a developer to specify a protocol in a high-level language and then compile the high level description into secure and efficient multiparty protocol."

The languages will allow developers to write high-level distributed applications for the secured multiparty computation while specifying what security properties they want to guarantee. HACCLE project research also will provide strong formal guarantees that the computations provide the desired functionality and security.

Provided by Purdue University

Citation: Protecting privacy at the ballot box with secure multiparty computation (2019, June 6) retrieved 29 April 2024 from https://phys.org/news/2019-06-privacy-ballot-multiparty.html