Designing Secure Hardware

SIMHA SETHUMADHAVAN

Assistant Professor of Computer Science

All computer software has one thing in common: it runs on computer hardware. But what if you could not trust the hardware to securely run software? That's the question posed by Simha Sethumadhavan. "If the hardware is hacked, then it can subvert all software and software security countermeasures,"said Sethumadhavan. "Since hardware is the root of trust, attacks on hardware are potentially very dangerous."

Until recently, computer scientists never suspected that someone could tamper with hardware. Yet investigators have found unusual additions in military chips. One way to prevent hardware hijackings is by passing tokens every time data moves within hardware. Sethumadhavan likens this to sending a thank you card after a gift.

"Let's say Charlie wants to contribute $100 to Alice's charity, but has to send it to Bob first," he said. "Bob takes $10 for himself and pays the rest to Alice. One way to find out if there is a problem is for Alice to write Charlie a thank you note for the $90 donation. When Charlie sees the discrepancy, he asks accountants to trace the missing money."

Sethumadhavan proposes creating similar triangle-like structures within a computer processor. "They would monitor any irregularities. We want to create a chain of monitored data and sound the alarm if any of the links break. These lightweight monitoring additions incur very little processing overhead," he said.

"We are taking a clean slate, ground-up approach to designing secure systems," he added. "As a foundational step, we have designed methods to protect processors, the core of all computing infrastructure. Once processors are secure, we can securely build out support for protecting other hardware and software."

Sethumadhavan is also working on other techniques for securing processors. "All hardware back doors have triggers and payloads. The triggers are usually time or data input values that activate the payload," he said.

"We are working on ways to silence the trigger," he said. "For example, we might be able to reset the processor's counter so it never reaches the threshold value needed to trigger an event. Or we could use lightweight encryption to obscure data values." Only when we fully trust our processor can we fully trust other security procedures, Sethumadhavan concluded.

Sethumadhavan is leading a project on rethinking security, making it a priority instead of an afterthought, with three other Columbia Engineering professors and a team from Princeton University. The project, titled "SPARCHS: Symbiotic, Polymorphic, Autotomic, Resilient, Clean-slate, Host Security," is funded by a federal grant for more than $6 million.

B.S.E., University of Madras, 2000; M.S., University of Texas, 2005; Ph.D., University of Texas, 2007