Secure Hardware Processors using Silicon Physical One-Way Functions

Dwaine Clarke, Blaise Gassend, Marten van Dijk and Srinivas Devadas

Introduction: *Physical one way functions* (POWF) are functions that combine an input value with the state of a physical system to produce an output value. In addition, they have in common with classical *one way functions* (OWF) that they are difficult to invert: given an output value, it is hard to find an input value and a physical system that would produce that output.

POWFs were introduced in [1], where they are implemented by shining a mobile laser beam through a non-homogenous medium and observing the resulting speckle pattern. They were used to make unclonable ID cards. Indeed, an important characteristic of POWFs is that when it is difficult to reproduce the physical system or to characterize it precisely enough to simulate it, an unclonable system results.

In many current applications, in particular smart cards, unclonability is provided by supplying a supposedly secure chip with a key that is supposed to remain hidden within the chip. The chip proves its identity by proving that it is the bearer of the right key. Unfortunately, experience shows that a host of techniques are available for hackers to extract the key from a chip (see [2]). Once the key is extracted an attacker knows everything about the chip and is able to make a clone, or a malicious imitation of it.

Approach: Our group is interested in using POWFs to get around the fragile security of secure chips that presently resides in being able to keep the bits of the key secret. To achieve our goal, we intend to use the manufacturing variations between chips to produce POWFs in silicon. We call these *silicon POWFs* (or SPOWFs). Our hope is that the variations in chip parameters will prove sufficient to identify a chip (previous work on identification can be found in [3]) as well as to produce a POWF that characterizes the chip in a secure way.

This project is currently at an early and exciting stage. So far, we have been trying to get a general idea of how to build and use SPOWFs. Once this is achieved, we can build a secure processor that bases its security on the POWF. Such a processor would answer three main problems:

- Identification: "Is this the chip I think it is?" This application is well suited to smartcards. The bearer of the card is given certain privileges, such as being able to withdraw money from his bank account.
- Certified execution: "Did my program get executed on the processor that I wanted it to execute on?" This could be of interest for anonymous computing where one person wants to execute a computation intensive application on the computer of someone he doesn't trust. If the result of the computation is accompanied by a certificate that proves that it was carried out by a certain chip, and the chip manufacturer recognizes that chip as being authentic, then he can consider that the computation was carried out correctly.
- Software protection: "I want this code to be executable only on a specific chip." This is one of the software industry's dreams. Algorithms could be encrypted so that they can only be executed on a single secure processor.

Research Support: This research was supported by Acer Inc., Delta Electronics Inc., HP Corp., NTT Inc., Nokia Research Center, and Philips Research under the MIT Project Oxygen partnership, and by DARPA through the Office of Naval Research under contract number N66001-99-2-891702.

References:

[1] P. S. Ravikanth, *Physical One-Way Functions*, Ph.D. thesis, Massachusetts Institute of Technology, March 2001.

[2] R. Anderson and M. Kuhn, "Tamper resistance - a cautionary note," in *proceedings of the Second Usenix Workshop on Electronic Commerce*, 1996, pp. 1–11.

[3] K. Lofstrom, W. R. Daasch, and D. Taylor, "IC identification circuit using device mismatch," in *2000 IEEE International Solid-State Circuits Conference*, 2000, pp. 372–373.