Hardware-Enforced Privacy

This installment of Computer’s series highlighting the work published in IEEE Computer Society journals comes from IEEE Computer Architecture Letters.

Simha Sethumadhavan, Columbia University

Privacy is important to modern, civilized life: it’s been described as a fundamental human right, a key to democratic processes, and—as evidenced by the rise of data brokers and markets—a driver of economic growth. Yet support for privacy isn’t considered a requirement in the design of hardware systems. In our paper “Hardware Enforced Statistical Privacy,” Matthew Maycock and I describe the motivation for and benefits of enforcing privacy in hardware (IEEE Computer Architecture Letters, vol. 15, no. 1, 2016, pp. 21–24).

The backdrop for our work is the current Internet of Things era. Today’s electronic devices have both computational power and the means to communicate not only with one another but also with remote third parties. Such devices’ increasing ubiquity has resulted in swaths of disparate user data that’s used to construct fine-grained behavioral profiles. These profiles can serve both noble causes, such as improving healthcare, and nefarious activities, such as illegal surveillance and targeting. As such, it’s vital that users have the tools necessary to control their privacy. Complicating privacy control, however, is the fact that users might want to give different-quality data to different parties, for example, doctors versus insurance companies or advertisers.

We argue that relegating such complex privacy needs to purely software solutions isn’t desirable. Software is mutable, generally buggier than hardware, might have coverage holes due to heterogeneity and layering, and might implement incorrect privacy notions. Hardware, however, is immutable and can sit between data sources (sensors) and data consumers (software accessing the data), guaranteeing coverage and a universal, minimum notion of privacy. Additionally, hardware combined with other trusted hardware mechanisms can provide proof of privacy—which isn’t easily achieved with software implementations alone.

We’ve proposed a hardware privacy protection unit (PPU): a standalone chip or an integrated hardware module that sits between the sensors and the remaining system (see Figure 1). The PPU can mediate data access, access sensor data, fuzz data, log overall transactions, and raise hardware exceptions for any issue at any step. The PPU can also be programmed with different policies. The type of fuzzing we studied was ε-differential privacy, a form of statistical privacy that involves noising the sensor outputs to provide strong mathematical guarantees.

Hardware-enforced privacy offers a cleaner way to engineer privacy into computing platforms. If successful, our approach could lead to a firmer foundation for reaping the rewards of the Internet of Things era.

SIMHA SETHUMADHAVAN is an associate professor of computer science at Columbia University. Contact him at simha@columbia.edu.

Figure 1. Hardware privacy protection unit. SoC: system on chip.