Invited Talk

Crypto Engineering: Some History and Some Case Studies (Extended Abstract)

Christof Paar
Chair for Communication Security
Electrical Engineering and Information Sciences Dept.
Ruhr-Universität Bochum
www.crypto.rub.de

Abstract of the Extended Abstract In this extended abstract, I will first try to describe briefly the developments in the cryptographic engineering community over the last decade. After this, some hopefully instructive case studies about cryptographic implementations in the real world will be given.

1 Timing the Market or: Why Did CHES Grow so Rapidly?

Exactly 10 years have passed since Cetin Koc and myself started the CHES (Cryptographic Hardware and Embedded Systems) workshop series. When the idea for the workshop was conceived in 1998 we had planned for a small, highly focused workshop in our favorite research area of cryptographic hardware, and we expected 50, perhaps 60, attendees. When almost 160 people showed up, Cetin and I knew that there was something special about this field. Even though this was a pleasant surprise, I had no idea how broad the area of cryptographic engineering would evolve in the years to come. In the following I would like to take the chance and speculate a bit about the reasons why CHES has grown to one of the most important events in applied cryptography.

At that time of the first CHES, my main interest and expertise was in hardware and software algorithms for asymmetric cryptography. Since public-key algorithms were very arithmetic intensive and both hardware and software performance was a far cry from what it is today, it was clear that much research was needed. Implementing RSA with an astronomically large modulus of 512 bit on 1990s PCs with acceptable run times was a major undertaking. Thus, at the time we started to plan CHES, the main focus in cryptographic engineering was on fast public-key implementation techniques, such as, for example [8, 5]. Even though there was certainly some work done on fast block cipher implementations (especially DES, but also IDEA and other ciphers), most of the scientifically challenging work targeted asymmetric implementations.

Thus, my view on the field where research in crypto engineering should take place was roughly described by the rather compact Table 1.

	HW impl.	SW impl.
asymmetric alg.	x	x

Table 1. My world view on crypto engineering, ca. 1998.

In the late 1990s, several developments took place which lead to an almost explosive growth of the area of cryptographic engineering. I see (at least) four main driving forces:

Side-Channel Attacks The notion of

$$\textit{Crypto Engineering} = \textit{Efficient Implementation}$$

started to change in the second half of the 1990s. In 1996, the Bell Core attack as one of the first fault injection attack was published [3]. In the next three years timing attacks, simple power analysis and differential power analysis were presented [9]. Not only the smart card industry was under shell shock, but the crypto implementation community realized very quickly that its formula had to be extended to:

$$\textit{Crypto Engineering} = \textit{Efficient Implementation} + \textit{Secure Implementation}$$

AES In 1997 the AES selection process had started. For the community of implementers, the AES competition became interesting in earnest in 1998/99, in other words, after the algorithms had been submitted and the first ciphers were excluded. This sparked an increase interest in the implementation aspects of symmetric-key algorithms.

Cryptology Research Matured Until the early 1990s, there were relatively few researchers working in cryptography outside government agencies. The field of cryptographic implementations was only a niche discipline with even fewer active people doing research. Publications were scattered over the cryptographic and engineering literature. The cryptographic community was well served by two flagship conferences, namely CRYPTO and EUROCRYPT, which were sufficient for reporting the most important developments in cryptology every year. However, the increasing number of researchers together with the increased understanding of many theoretical and practical issues in cryptology triggered a specialization and extension of the crypto conference landscape. Starting with FSE (Fast Software Encryption) in 1993, PKC (Public-Key Cryptography) in 1998 and CHES in 1999, several new workshops in sub-areas of cryptography served the need of a larger and more specialized scientific community. I believe this is the natural and healthy evolution of a discipline which is maturing.

Internet Boom The last development which helped to push CHES and the field of crypto engineering was the dot-com boom in the late 1990s. There was both a perceived and a real need for everything that was related to information technology. Applied cryptography was considered part of the whole

brave new world of the Internet area, and many companies started or enlarged their security groups. As part of that development, crypto engineering was also receiving more attention.

All of these factors contributed to extend the scope of CHES considerably. Within three years, there were more than 200 attendees and more than 100 submissions. Hence in hindsight, the reason why CHES has become such an important conference was there was almost a perfect *market timing* for starting CHES in 1999: the time was simply ripe for such an event.

In the years since then, new topics such as lightweight crypto, true random number generators (TRNG), cryptanalytical hardware and physical unclonable functions (PUF) were also added to the repertoire of topics treated at CHES. Thus, a current listing of the sub-areas of modern crypto engineering is more accurately described by this table:

	HW impl.		SW impl.		Secure impl.		TRNG	cryptanal.	PUF
	lightweight	high speed	lightweight	high speed	passive	fault inj.		HW	
asymmetric	x	x	x	x	x	x	x	x	x
symmetric	x	x	x	x	x	x			

Table 2. The field of crypto engineering in 2009.

The table should certainly not be taken as the final verdict on the spectrum of topics within crypto engineering. For instance, new topics like Trojan hardware (as evident by the Hot Topics Session of this year's CHES), are emerging and should certainly be included.

2 Embedded Cryptography in the Wild: Some Case Studies

Cryptography has sneaked into everything, from web browsers and email programs to cell phones, bank cards, cars and even into medical devices. In the near future we will find many new exciting applications for cryptography such as RFID tags for anti-counterfeiting or car-to-car communications. I want to briefly mention research projects we have been involved in which cryptography was instrumental for securing new embedded applications.

Lightweight Cryptography for RFID Tags and such With the advent of pervasive computing, an increasing demand to integrate cryptographic functions in applications has risen. Different from the past, it is often desirable to have cryptographic primitives that are as small as possible. There are two main reasons for this. First, there are applications constrained by a hard limit with respect to gate count or power. The prime example are RFID tags on which it is simply physically impossible to implement RSA-1024. The second reason are applications which are heavily cost constrained, e.g., high-volume consumer devices.

Here it would be possible to integrate non-optimized crypto engines but it is highly desirable to use implementations which cause the smallest possible cost increase for the product.

With this goal in mind, a team of researchers developed the PRESENT block cipher [2]. It can be implemented with as little as 1000 gate equivalences [11] which is close to the theoretical limit if one has to store 64 state bits and 80 key bits. For the asymmetric case, we developed an extremely small elliptic curve engine which requires between roughly 10,000 and 15,000 gate equivalences, depending on the security level [10].

High Performance Elliptic Curve Engine for Car-to-Car Communication Air pollution is not the only health hazard posed by cars. They are also quite deadly when it comes to accidents. In the developed world, traffic fatalities are, by a far margin, the most common cause of death caused by accidents. Both in the European Union and in the USA there are more than 40,000 traffic fatalities annually, and world-wide they are the leading cause of death for people in the age range of 10–24. Given that many mechanically safety measure such as seat belts, air bags and anti-block brake (ABS) systems are very far advanced, there has been a push in the last few years to develop electronic driver assistant systems. One major motivation is to reduce the number of accidents. Some driver assistant systems are based on car-to-car (C2C) and car-to-infrastructure (C2I) communication. If such systems were in place, many collisions between vehicles could be avoided. One requirement of C2C and C2I systems is that the communication should be secure. It does not take much fantasy to imagine how an attacker could cause quite serious trouble if, for instance, faked collision warning messages are issued to cars driving on the German autobahn with 200 km/h.

The IEEE Standard 1609 calls for a digital signature over every position message sent out by every car. In a high-density traffic environment that could translate in 1000 or more digital signatures which have to be verified per second. The challenge here is to develop an ECC engine that can support thousands of verifications per second at affordable costs. We developed new ECC engines that make use of the DSP cores available on modern FPGAs. Our engine can verify more than 2000 ECC signatures (224 bit NIST curve) with a mid-size commercial FPGA [6]. Previously such speeds were only achievable with expensive and power-consuming parallel CPUs or with ASICs. Our design scales theoretically to more than 30,000 signatures per second on high-end FPGAs.

Side-Channel Attacks against Remote Keyless Entry Systems Ever since side-channel analysis (SCA) were proposed (cf. the first section of this abstract) it was recognized that they pose a major risk for real-world systems. There had been many anecdotal reports, especially from the smart card industry, about the vulnerability against SCA. However, despite hundreds of research papers in this area, there had been hardly any descriptions of an SCA against an actual system.

Last year we attacked the KeeLoq remote keyless entry system using SCA. KeeLoq was an instructive target. It is a 1980s symmetric cipher against which several analytical attack had been proposed in short sequence [1, 4, 7]. However, due to the mode of operation of keyless entry systems, the required plaintext-ciphertext pairs are almost impossible to obtain in the real world. In contrast, using a DPA-like attack, we showed that both the individual transmitter keys (which are typically embedded in garage door openers or car keys) as well as system-wide manufacturer keys can be extracted. Once the manufacturer key has been recovered after a few thousands measurements, transmitters can be cloned after simply eavesdropping on one or two communications.

References

1. A. Bogdanov. Attacks on the KeeLoq Block Cipher and Authentication Systems. In *3rd Conference on RFID Security 2007 (RFIDSec 2007)*. `http://rfidsec07.etsit.uma.es/slides/papers/paper-22.pdf`.
2. A. Bogdanov, G. Leander, L. R. Knudsen, C. Paar, A. Poschmann, M. J. Robshaw, Y. Seurin, and C. Vikkelsoe. PRESENT - An Ultra-Lightweight Block Cipher. In *Proceedings of CHES 2007*, number 4727 in LNCS, pages 450 – 466. Springer.
3. D. Boneh, R. A. DeMillo, and R. J. Lipton. On the importance of checking computations, 1996. `http://citeseer.ist.psu.edu/491209.html`.
4. N. T. Courtois, G. V. Bard, and D. Wagner. Algebraic and Slide Attacks on KeeLoq. In *Fast Software Encryption - FSE 2008*, Lecture Notes in Computer Science. Springer, 2008.
5. S. E. Eldridge and C. D. Walter. Hardware implementation of Montgomery's modular multiplication algorithm. *IEEE Transactions on Computers*, 42(6):693–699, July 1993.
6. T. Güneysu and C. Paar. Ultra High Performance ECC over NIST Primes on Commercial FPGAs. In *CHES '08: Proceeding sof the 10th international workshop on Cryptographic Hardware and Embedded Systems*, pages 62–78, Berlin, Heidelberg, 2008. Springer-Verlag.
7. S. Indesteege, N. Keller, O. Dunkelman, E. Biham, and B. Preneel. A Practical Attack on KeeLoq. In *Advances in Cryptology - EUROCRYPT 2008*, Lecture Notes in Computer Science. Springer, 2008.
8. C. K. Koc, T. Acar, and J. Burton S. Kaliski. Analyzing and comparing montgomery multiplication algorithms. *IEEE Micro*, 16(3):26–33, 1996.
9. P. Kocher, J. Jaffe, and B. Jun. *Differential Power Analysis*, volume 1666. 1999.
10. S. Kumar. *Elliptic Curve Cryptography for Constrained Devices*. PhD thesis, Electrical Engineering and Information Sciences Department, Ruhr-University of Bochum, 2006.
11. C. Rolfes, A. Poschmann, G. Leander, and C. Paar. Ultra-Lightweight Implementations for Smart Devices-Security for 1000 Gate Equivalents. In *Proceedings of the 8th International Conference on Smart Card Research and Advanced Applications (CARDIS 2008)*, volume 5189, pages 89–103. Springer, 2008.