STSM Project Report
Cost Action IC1204 STSM

Peter Schwabe

December 6, 2013

Host Institution:	TU Graz, Institute for Applied Information Processing and Communications, Dr. Michael Hutter
Visitor:	Dr. Peter Schwabe, University Nijmegen, Digital Security Group
Period:	The visiting period was from Monday, October 28, 2013 until Wednesday, November 6, 2013 (9 days).

Planned Research

The main focus of the planned research was to consider multiplication algorithms with *subquadratic* complexity on the AVR microcontroller. In particular we planned to investigate possible speedups from the Karatsuba multiplication [4] technique on AVR microcontrollers for input sizes that are typically used in the context of elliptic-curve cryptography.

Fast multiplication on the AVR is one building block in a larger project that aims at bringing side-channel-protected fast cryptography to AVR microcontrollers.

Currently we have protection only against timing attacks. The next step will be to implement countermeasures against power-analysis attacks. For the elliptic-curve arithmetic we are planning to use countermeasures that make use of the algebraic structure of elliptic curves (e.g., projective randomization, scalar randomization, and curve isomorphisms). To protect symmetric cryptography we will investigate efficient masking and hiding schemes. A longer-term goal is to also investigate protection against fault-injection attacks.

Layout of the work during the visit

The scientific visit in Graz produced software for big-integer multiplication on AVR microcontrollers that beats the current speed record presented in [3] and [5] for inputs sizes between 80 and 192 bits.

Another work item that we put on our agenda during the visit was to bring the software that we presented in [2] to an ATmega Smartcard. The reason is that we can then use established side-channel-attack setups for smartcards in both the group in Graz and the group in Nijmegen to investigate power-analysis attacks and countermeasures.

Bringing the NaCl library to the smartcard required rewriting some parts of the code to reduce RAM usage and, more importantly, to implement the T1 communication protocol on that smartcard. We now have a full Curve25519 [1] implementation running on the smartcard including communication with a host PC.

Detailed description of work

A large part of the work was dedicated to investigating which form of the Karatsuba equation yields better results. The "standard" way to express the idea of Karatsuba algorithm is to decompose multiplication of two $2m$-bit integers $A = (a_0 + 2^m a_1)$ and $B = (b_0 + 2^m b_1)$ as

$$AB = a_0b_0 + ((a_0 + a_1)(b_0 + b_1) - a_0b_0 - a_1b_1)2^m + a_1b_12^{2m}.$$

A slightly different way to express the same idea is the equation

$$AB = a_0b_0 + ((a_0 - a_1)(b_0 - b_1) + a_0b_0 + a_1b_1)2^m + a_1b_12^{2m}.$$

From a complexity-theory point of view it does not matter which of the two equations is used; recursive application of the decomposition yields complexity $\Theta(n^{\log_2 3})$ for the multiplication of n-bit integers. However, for implementations and performance measured in clock cycles instead of asymptotic complexity it does make significant difference. The first equation may produce carry bits in the addition of $a_0 + a_1$ and $b_0 + b_1$; multiplication of these two values needs to also handle these carry bits. The second equation has the trouble, that subtraction of $a_0 - a_1$ and $b_0 - b_1$ may yield a negative result, so we have to distinguish different cases (depending on the sign of these values). This is not only a problem for performance, but also for side-channel security: Distinguishing between different cases through branching may create vulnerabilities against timing attacks. We achieve the current new speed records using the first equation but we expect even further speedups using the second equation – even for constant-time code that is fully protected against timing attacks.

The second work item (bringing NaCl to the ATmega card) was not as much scientific work as an engineering task that was required for future research. The current implementation is functional for the side-channel-attack experiments that we are planning to carry out (for example, DPA against static Diffie-Hellman and horizontal correlation analysis). We are planning to release a more extensive version of the smartcard software, together with documentation of how to create your own "NaClcard" into the public domain.

Impression of the visit

The 8 days in Graz were highly productive and I very much appreciated the environment offered by TU Graz for this visit. I got an office assigned where I could work on my own when I wanted to, but most of the time I worked together with Michael in his office. The joint group lunches gave my the opportunity to also informally discuss with other people in the group. The visit was definitely very helpful in bringing the Digital Security Group of RU Nijmegen and the IAIK at TU Graz closer together and propagate further collaboration.

Future work

The current software for Karatsuba multiplication only uses one level of the recursion; multiplication of smaller integers is handled by standard schoolbook-like methods. To extend the results to larger integers we will use multiple levels of the Karatsuba recursion. In particular, we are interested in speeding up 256-bit multiplication to speed up the Curve25519 computation in NaCl for AVR.

With the smartcard running Curve25519 we can proceed to mount power-analysis attacks against this software, which, so far, is protected only against timing attacks. This work is collaborative work with Lejla Batina (RU Nijmegen), Louiza Papachristodoulou (RU Nijmegen), and Ingo von Maurich (Ruhr Universität Bochum).

Planned publications

We are planning to submit the results of the research on big-integer multiplication on AVR to CHES 2014.

Research on attacking and defending Curve25519 on the ATmega card is still in an early stage so it is hard to predict where we will submit a paper resulting from this research. We expect publishable results by mid 2014 and will then choose a suitable venue to submit to.

References

[1] Daniel J. Bernstein. Curve25519: new Diffie-Hellman speed records. In Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malkin, editors, *Public Key Cryptography – PKC 2006*, volume 3958 of *Lecture Notes in Computer Science*, pages 207–228. Springer-Verlag Berlin Heidelberg, 2006. Document ID: 4230efdfa673480fc079449d90f322c0, http://cr.yp.to/papers.html#curve25519. 1

[2] Michael Hutter and Peter Schwabe. Nacl on 8-bit avr microcontrollers. In Amr Youssef and Abderrahmane Nitaj, editors, *Progress in Cryptology – AFRICACRYPT 2013*, volume 7918 of *Lecture Notes in Computer Science*, pages 156–172. Springer-Verlag Berlin Heidelberg, 2013. Document ID: cd4aad485407c33ece17e509622eb554, http://cryptojedi.org/papers/#avrnacl. 1

[3] Michael Hutter and Erich Wenger. Fast multi-precision multiplication for public-key cryptography on embedded microprocessors. In Bart Preneel and Tsuyoshi Takagi, editors, *Cryptographic Hardware and Embedded Systems – CHES 2011, 13th International Workshop, Nara, Japan, September 28 – October 1, 2011. Proceedings*, volume 6917 of *Lecture Notes in Computer Science*, pages 459–474. Springer-Verlag Berlin Heidelberg, 2011. https://online.tugraz.at/tug_online/voe_main2.getvolltext?pCurrPk=58138. 1

[4] Anatolii Karatsuba and Yuri Ofman. Multiplication of multidigit numbers on automata. *Soviet Physics Doklady*, 7:595–596, 1963. Translated from Doklady Akademii Nauk SSSR, Vol. 145, No. 2, pp. 293–294, July 1962. 1

[5] Hwajeong Seo and Howon Kim. Multi-precision multiplication for public-key cryptography on embedded microprocessors. In Dong Hoon Lee MotiYung, editor, *Information Security Applications*, volume 7690 of *Lecture Notes in Computer Science*, pages 55–67. Springer-Verlag Berlin Heidelberg, 2012. http://isaa.sch.ac.kr/wisa2012/%EB%85%BC%EB%AC%B8/Session%202/1-130_Multi-precision%20Multiplication%20for%20Public-Key%20Cryptography%20on%20Embedded%20Microprocessors.pdf. 1