Fast Skinny-128 SIMD Implementations for Sequential Modes of Operation

Proposal for presentation at NIST LWC 2022

Alexandre Adomnicai[1], Kazuhiko Minematsu[2] and Maki Shigeri[3]

[1] CryptoNext Security, Paris, France
[2] NEC Corporation, Kawasaki, Japan
[3] NEC Solution Innovators, Hokuriku, Japan

While the primary goal of the NIST LWC project is to select standards for efficient authenticated encryption on resource-constrained devices (e.g. low-cost microcontrollers), these algorithms will inevitably be deployed on more sophisticated platforms (e.g. smartphones, servers) for interoperability purposes. Although such platforms show less computational limitations, having dedicated efficient implementations is a nice-to-have feature since it may have to communicate with many devices simultaneously. When high parallelism can be achieved in the operating mode where the iternal cryptographic primitive will be placed, one can always use highly bitsliced implementations that can lead to excellent performance. Actually, the best software results reported for Skinny-128 on Intel SIMD are obtained by processing 64 128-bit blocks (i.e. 1KiB) of data in parallel [BJK+16]. However, such highly bitsliced implementations are not relevant when processing small payloads or for sequential (i.e. non-parallelizable) operating modes as used in Romulus [IKMP20], one of the 10 NIST LWC finalists. In this talk, we introduce an implementation strategy to optimize the performance of Skinny-128 for sequential modes of operation on SIMD platforms. Our main optimization trick consists in decomposing the 8-bit S-box into smaller ones so that we can take advantage of SIMD-specific vector permute instructions to reach competitive performance without introducing secret-dependent timing variations. We applied our implementation strategy to all Romulus variants on ARM Neon and Intel SSE processors. As a result, we observe a speedup by a factor that ranges from 1.5 to 4.5 depending on the computing platform, compared to fixsliced implementations [AP20], which constitute the current best constant-time option when processing blocks in a sequential manner. Another benefit of our implementations is the memory footprint, since the stack consumption is reduced up to a factor of 5.

References

[AP20] Alexandre Adomnicai and Thomas Peyrin. Fixslicing AES-like Ciphers: New bitsliced AES speed records on ARM-Cortex M and RISC-V. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2021(1):402–425, Dec. 2020.

[BJK+16] Christof Beierle, Jérémy Jean, Stefan Kölbl, Gregor Leander, Amir Moradi, Thomas Peyrin, Yu Sasaki, Pascal Sasdrich, and Siang Meng Sim. The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS. In *CRYPTO (2)*, volume 9815 of *Lecture Notes in Computer Science*, pages 123–153. Springer, 2016.

[IKMP20] Tetsu Iwata, Mustafa Khairallah, Kazuhiko Minematsu, and Thomas Peyrin. Duel of the Titans: The Romulus and Remus Families of Lightweight AEAD Algorithms. *IACR Transactions on Symmetric Cryptology*, 2020(1):43–120, May 2020.