Updates on Elephant
(proposal for presentation)

Tim Beyne[1], Yu Long Chen[1], Christoph Dobraunig[2,3], and Bart Mennink[2]

[1] KU Leuven and imec-COSIC, Leuven, Belgium
[2] Radboud University, Nijmegen, The Netherlands
[3] Graz University of Technology, Austria
elephant@cs.ru.nl

Elephant is a nonce-based encrypt-then-MAC style authenticated encryption scheme. It is permutation-based and only evaluates this permutation in the forward direction. It is parallelizable by design, and as such perfectly suitable for small permutations. The Elephant scheme consists of three instances: Dumbo, Jumbo, and Delirium, which are instantiations of Elephant with Spongent-π[160], Spongent-π[176], and Keccak-f[200], respectively.

In this talk, we will discuss a tweak we are planning to apply to the Elephant mode. In a nutshell, the main change consists of moving from a Wegman-Carter-Shoup style authenticator [3,6,7] in v1.1 to a protected counter sum style authenticator [2,5] in v2. Elephant v2 is depicted in Figure 1. We will explain that this planned tweak does not degrade the security and efficiency of Elephant. In addition, we will discuss the main benefit of this tweak: whereas version v1.1 only achieved confidentiality and authenticity against nonce-respecting adversaries, v2 additionally achieves *authenticity under nonce-reuse*.

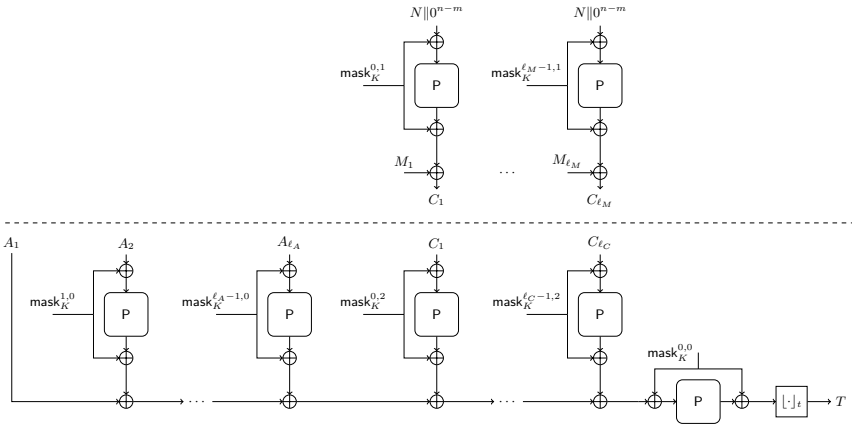

Fig. 1. Depiction of Elephant. For the encryption part (top): message is padded as $M_1 \dots M_{\ell_M} \xleftarrow{n} M$, and ciphertext equals $C = \lfloor C_1 \dots C_{\ell_M} \rfloor_{|M|}$. For the authentication part (bottom): nonce and associated data are padded as $A_1 \dots A_{\ell_A} \xleftarrow{n} N\|A\|1$, and ciphertext is padded as $C_1 \dots C_{\ell_C} \xleftarrow{n} C\|1$.

In addition, we will discuss novel results on the implementation of Elephant. We will consider our own[4] as well as external [1, 4] results on using the inherent parallelism of Elephant.

References

1. Belaïd, S., Dagand, P., Mercadier, D., Rivain, M., Wintersdorff, R.: Tornado: Automatic Generation of Probing-Secure Masked Bitsliced Implementations. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 311–341. Springer (2020)
2. Bernstein, D.J.: How to Stretch Random Functions: The Security of Protected Counter Sums. J. Cryptology 12(3), 185–192 (1999)
3. Bernstein, D.J.: Stronger Security Bounds for Wegman-Carter-Shoup Authenticators. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 164–180. Springer (2005)
4. Campos, F., Jellema, L., Lemmen, M., Müller, L., Sprenkels, D., Viguier, B.: Assembly or Optimized C for Lightweight Cryptography on RISC-V? Cryptology ePrint Archive, Report 2020/836 (2020)
5. Luykx, A., Preneel, B., Tischhauser, E., Yasuda, K.: A MAC Mode for Lightweight Block Ciphers. In: Peyrin, T. (ed.) FSE 2016. LNCS, vol. 9783, pp. 43–59. Springer (2016)
6. Shoup, V.: On Fast and Provably Secure Message Authentication Based on Universal Hashing. In: Koblitz, N. (ed.) CRYPTO '96. LNCS, vol. 1109, pp. 313–328. Springer (1996)
7. Wegman, M.N., Carter, L.: New Hash Functions and Their Use in Authentication and Set Equality. J. Comput. Syst. Sci. 22(3), 265–279 (1981)

[4] Available at `https://github.com/TimBeyne/Elephant`.