Updates on the Subterranean 2.0 cipher suite

Joan Daemen, Pedro Maat Costa Massolino, Alireza Mehrdad and Yann Rotella

Radboud University, Digital Security Department, Nijmegen, The Netherlands

Date: September 18, 2020

In this note we report on some new results related to Subterranean since March 2019, when round 1 of the NIST lightweight competition started, namely:

- improved security analysis and design rationale,
- new hardware implementations,
- new third-party cryptanalysis,
- low multiplicative complexity aspect.

1 Improved security analysis and design rationale

In the special issue dedicated to the lightweight competition at Transactions on Symmetric Cryptology in June 2020, we, the designers of Subterranean 2.0 and Alireza Mehrdad, published a paper on the design and analysis of the Subterranean 2.0 cipher [2].

- In this paper we improved the write-up of the Subterranean design rationale as compared to our original submission to NIST. We added the description of round function properties leading to stronger and more rigourous arguments on the design choices and on the security analysis.
- Additionally, the paper provides lower bounds for the weight of differential trails for all number of rounds, from 1 to 8. These bounds are the result of a considerable effort by Alireza Mehrdad, also affiliated at Radboud University Nijmegen. Alireza is now a member of the Subterranean team.

2 Hardware implementation results

Our paper at ToSC [2] included new hardware implementation results. Since then, we have made an implementation that is compliant to the LWC Hardware API [3].

Figure 1 shows the Subterranean circuit that is compatible with the LWC Hardware API. The circuit was entirely made in Verilog and does not use the provided LWC Hardware API components in VHDL. By making our custom interface we can focus on the required operations for Subterranean 2.0 suite, however our approach is very similar to the one used in LWC Hardware API components in VHDL. Our circuit has two input buffers one for each interface, the public and secret one, one output buffer and the main circuit that we call Subterranean stream. Subterranean stream is a circuit that can perform AEAD and hash of Subterranean 2.0 as long as data arrives in specific 32-bit blocks and the last block is explicitly flagged.

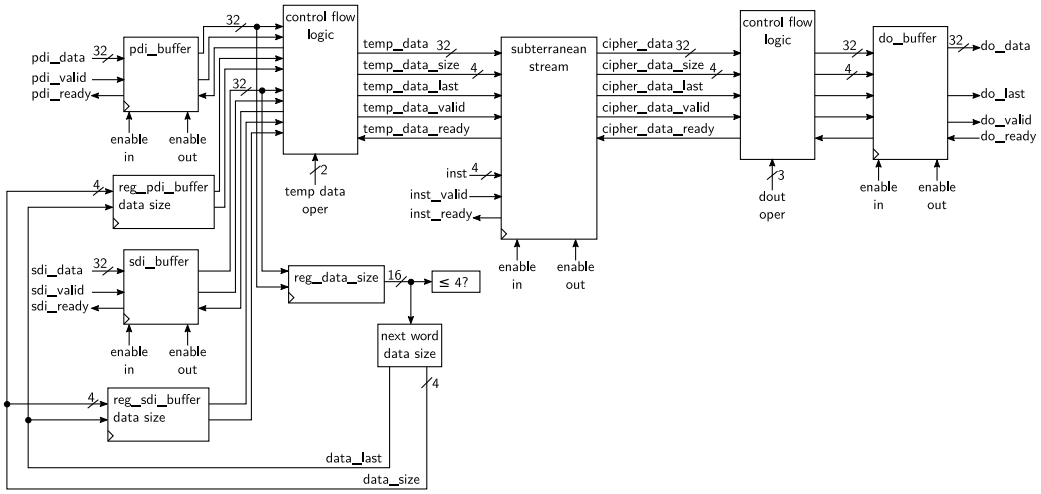

Figure 1: Subterranean 2.0 suite circuit compliant witht the LWC hardware API.

3 Third-party cryptanalysis

Since the beginning of the Lightweight Competition, some well-known cryptographers tackled the challenge to evaluate the security of Subterranean. In 2019, Fukang Liu, Takanori Isobe and Willi Meier published a paper titled "Cube-Based Cryptanalysis of Subterranean-SAE" at Transactions on Symmetric Cryptology [4]. While we did not design Subterranean to be resistant in a nonce-misuse scenario, those authors provided the first attack in a nonce-misuse scenario, that requires 2^{15} bytes of data. Additionally, they achieved to mount a key-recovery attack on a weakened version of Subterranean in a nonce-respecting scenario, where the weakening is a reduction of the number of blank rounds from 8 to 4. This last attack requires 2^{122} calls to the permutation and $2^{71.5}$ bytes of data. They also built a distinguisher for the 4-round permutation with a cost of 2^{33} calls, and the same number of blocks. Eventually, this cryptanalysis shows that the security margin is still high for the full Subterranean 2.0 cipher suite.

Moreover, we are also aware of another team of cryptographers that are working on a security analysis of Subterranean, further reinforcing the confidence in Subterranean. However, nothing has been published or made public yet.

4 Low multiplicative complexity

While reading [1], we realized that Subterranean has a low number of multiplications in GF(2) (or equivalently, binary AND gates) per encrypted bit. To the best of our knowledge, Subterranean has the lowest multiplicative complexity for long messages of all the NIST lightweight candidates, namely 8 per encrypted bit.

This is an interesting feature when one applies masking with many shares as discussed in [1] or for usage in so-called "*advanced cryptography*" such as multi-party computation and homomorphic encryption.

References

[1] Sonia Belaïd, Pierre-Évariste Dagand, Darius Mercadier, Matthieu Rivain, and Raphaël Wintersdorff, *Tornado: Automatic generation of probing-secure masked bitsliced implementations*, Advances in Cryptology - EUROCRYPT 2020 - 39th Annual International

Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, May 10-14, 2020, Proceedings, Part III (Anne Canteaut and Yuval Ishai, eds.), Lecture Notes in Computer Science, vol. 12107, Springer, 2020, pp. 311–341.

[2] Joan Daemen, Pedro Maat Costa Massolino, Alireza Mehrdad, and Yann Rotella, *The subterranean 2.0 cipher suite*, IACR Trans. Symmetric Cryptol. **2020** (2020), no. S1, 262–294.

[3] E. Homsirikamol, W. Diehl, A. Ferozpuri, F. Farahmand, M. U. Sharif, and K. Gaj, *A universal hardware api for authenticated ciphers*, 2015 International Conference on ReConFigurable Computing and FPGAs (ReConFig), 2015, pp. 1–8.

[4] Fukang Liu, Takanori Isobe, and Willi Meier, *Cube-based cryptanalysis of subterranean-sae*, IACR Trans. Symmetric Cryptol. **2019** (2019), no. 4, 192–222.