Updates on Spix

Riham AlTawy[1], Guang Gong[2], Morgan He[2], Kalikinkar Mandal[3], and Raghvendra Rohit[4]

[1] Department of Electrical and Computer Engineering, University of Victoria, Victoria, Canada
[2] Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Canada
[3] Faculty of Computer Science, University of New Brunswick, Fredericton, Canada
[4] University of Rennes, CNRS, IRISA, France

Abstract. Spix is an authenticated encryption algorithm that supports both messages and associated data (AD) and a round 2 candidate of the NIST lightweight cryptography standardization competition [3]. On the top level, Spix adopts the monkey duplex construction which operates on two instances of the 256-bit sLiSCP-light permutation. In this report, we provide updates on Spix since its selection as a round 2 candidate. In particular, we report third party cryptanalysis on its underlying permutation sLiSCP-light and new implementation results. We also highlight the target applications and our plans for future tweaks.

1 Security Analysis

Spix uses a 128-bit key and utilizes sLiSCP-light with state size 256 bits. An in-depth security analysis of Spix has already been provided in [1, 2]. Spix offers 128-bit security against attacks targeting confidentiality and integrity while the prescribed data limit is of 2^{60} bits in a nonce-respecting setting [1, Table 2.1].

At the time of the writing of this report, we are not aware of any third party attacks on Spix. However, there have been some presented distinguishers on reduced round variants of its underlying sLiSCP-light permutation and results on its leakage resistance which we briefly discuss below.

1.1 Distinguishers on round-reduced sLiSCP-light-256 permutation

Hosoyamada *et al.* have analyzed sLiSCP-light against limited-birthday distinguishers (LBD) which can cover up to 16 out of 18 rounds of the permutation [7]. The LBDs complexities are $2^{154.6}$ (time) and $2^{48.3}$ (memory) for sLiSCP-light-256. Although this work improves the existing distinguishers on sLiSCP-light-256 [2] by 2 rounds, they do not pose a direct threat to Spix. This is because the available degrees of freedom are 64 bits, i.e., rate width, per permutation call, while the proposed LBD requires much higher time complexity.

Kravela *et al.* have presented a differential path for 6 out of 18 rounds for sLiSCP-light-256 with probability $2^{-106.14}$ [8]. We emphasize that a similar analysis of differential trails has been investigated in detail by us in [2, 10]. More precisely, using MILP, we have presented differential trails with six active sboxes and bounded (not tight) its maximum expected differential characteristic probability by $(2^{-15.9})^6 = 2^{-95.4}$. We have also mentioned that we expect that a tighter bound for sLiSCP-light-256 exists because given the iterated nature of the employed sbox, long trails are preserved. Accordingly, the long trail strategy offers a better security argument than simply counting the minimum number of active sboxes [5]. To this end, the presented distinguisher in [8] strengthens our security claims.

The same authors further mentioned that *"the bits rc_1^{8+n} and rc_1^n are equal ... "*. We would like to point out this is a property of a primitive polynomial of LFSR and as such **it does not add anything new to the security**. The constants were chosen so that: 1) they can be generated by a single LFSR, and 2) **the pair of round constants and step constants at each round and step are distinct**. See [1, Section 3.1.4] for further analysis.

1.2 Leakage resistance of Spix

Bellizia *et al.* have analyzed the leakage-resistance of NIST LWC round 2 candidates [4]. Spix has been identified as CIML2 and CCAmL1 secure where CIML2 refers to "*ciphertext integrity with misuse-resistance (i.e., no constraint on nonces) and leakage in encryption and decryption*", while CCAmL1 denotes "*chosen* *ciphertext security with nonce misuse-resilience (i.e., fresh challenge nonce) and leakage in encryption only*". Further details are found in [4].

In summary, the third party cryptanalysis results on sLiSCP-light-256 do not affect the security claims of Spix.

2 Implementation Results

2.1 Software

The codes for the bitslice implementation of Spix and the C implementation of sLiSCP-light-256 using SSE2 and AVX2 instruction sets are available publicly on https://uwaterloo.ca/communications-security-lab/lwc/spix.

Implementation and benchmarking by Weatherley [11]. Spix is included in Rhy's Weatherley software benchmarking framework where the performance of round 2 candidates is evaluated on 8-bit and 32-bit microcontrollers. Further, a masked implementation of Spix for up to 6 shares has been incorporated in the same framework. For the details on the performance, the reader is referred to [11].

2.2 Hardware

We have provided the implementation of Spix in two technologies: ST Micro 65 nm and IBM 130 nm. The codes are available at https://uwaterloo.ca/communications-security-lab/lwc/spix. Note that our codes currently do not follow the LWC API framework of [9] and the API compliant hardware codes will be available shortly.

3 Features

Applications and use cases. Given that Spix is a hardware-oriented lightweight authenticated cipher that is primarily designed to achieve low *area, power and energy*, it fits the requirements of RFID and sensor network applications. Moreover, we have investigated Spix when used to implement the key derivation function (KDF) and generate the message integrity check (MIC) in IEEE 802.11X [6] and CoAP [13] handshake mutual authentication and key establishment protocols for IoT applications [12]. Given that the majority of IoT devices are quipped with microcontrollers, we provide performance evaluation of Spix for KDF and MIC functionalities and handshaking and data protection protocols on microcontrollers. Our experimental results show that Spix take about 3,176, 2,912, and 2,831 ms to complete the IEEE802.11X authentication protocol using ATmega128, MSP430F2370, and Cortex-M3, respectively. For the data protection protocol, Spix achieves a throughput of 9, 22 and 109 Kbits/s on ATmega128, MSP430F2370, and Cortex-M3, respectively to encrypt and authenticate a plaintext of 1024 bits and an associated data of 128 bits. More details can be found in [12].

Comparison with AES-GCM. A fair comparison of Spix with AES-GCM is hard unless both ciphers are implemented in the same technology. However, in ASICs, we believe that Spix outperforms AES-GCM in *area*. This is because AES-GCM requires a 128-bit finite field multiplier which is generally expensive in ASICs.

4 Proposed Tweaks

We do not plan to propose any tweaks in the design of Spix.

References

[1] Riham AlTawy, Guang Gong, Morgan He, Kalikinkar Mandal, and Raghvendra Rohit. SPIX: An authenticated cipher. Submission to NIST-LWC, 2019. https://csrc.nist.gov/CSRC/media/Projects/lightweight-cryptography/documents/round-2/spec-doc-rnd2/spix-spec-round2.pdf.

[2] Riham Altawy, Raghvendra Rohit, Morgan He, Kalikinkar Mandal, Gangqiang Yang, and Guang Gong. Sliscp-light: Towards hardware optimized sponge-specific cryptographic permutations. *ACM Transactions on Embedded Computing Systems (TECS)*, 17(4):81, 2018.

[3] Lawrence Bassham, Cagdas Calik, Donghoon Chang, Jinkeon Kang, Kerry McKay, and Meltem Sonmez Turan. Lightweight cryptography: Round 2 candidates, 2019. https://csrc.nist.gov/Projects/lightweight-cryptography/round-2-candidates.

[4] Davide Bellizia, Olivier Bronchain, Gaëtan Cassiers, Vincent Grosso, Chun Guo, Charles Momin, Olivier Pereira, Thomas Peters, and François-Xavier Standaert. Mode-level vs. implementation-level physical security in symmetric cryptography a practical guide through the leakage-resistance jungle. In *Crypto*, 2020.

[5] Daniel Dinu, Léo Perrin, Aleksei Udovenko, Vesselin Velichkov, Johann Großschädl, and Alex Biryukov. Design strategies for arx with provable bounds: Sparx and lax. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT*, pages 484–513. Springer, 2016.

[6] IEEE 802.11 Working Group et al. 802.11ax - ieee draft standard for information technology – telecommunications and information exchange between systems local and metropolitan area networks. *IEEE Std*, 2019.

[7] Akinori Hosoyamada, María Naya-Plasencia, and Yu Sasaki. Improved attacks on sliscp permutation and tight bound of limited birthday distinguishers. Cryptology ePrint Archive, Report 2020/1089, 2020. https://eprint.iacr.org/2020/1089.

[8] Liliya Kraleva, Raluca Posteuca, and Vincent Rijmen. Cryptanalysis of the permutation based algorithm spoc. Cryptology ePrint Archive, Report 2020/1072, 2020. https://eprint.iacr.org/2020/1072.

[9] Ekawat Homsirikamol William Diehl Jens-Peter Kaps Michael Tempelmeier, Farnoud Farahmand and Kris Gaj. Implementer's guide to hardware implementations compliant with the hardware api for lightweight cryptography, v1.0.1 (nov 2019). https://cryptography.gmu.edu/athena/LWC/LWC_HW_Implementers_Guide.pdf.

[10] Rohit, Raghvendra. Design and cryptanalysis of lightweight symmetric key primitives, 2020.

[11] Rhys Weatherley. https://rweather.github.io/lightweight-crypto/.

[12] Yunjie Yi, Guang Gong, and Kalikinkar Mandal. Implementation of lightweight ciphers and their integration into entity authentication with IEEE 802.11 physical layer transmission, 2020. In submission at IEEE Internet of Things Journal.

[13] K. Hartke C. Bormann Z. Shelby, K. Hartke. The constrained application protocol (coap). RFC 7252, 2014. https://tools.ietf.org/pdf/rfc7252.pdf.