NIST Update: ISAP v2.0

Christoph Dobraunig, Maria Eichlseder, Stefan Mangard, Florian Mendel,
Bart Mennink, Robert Primas and Thomas Unterluggauer

https://isap.iaik.tugraz.at
isap@iaik.tugraz.at

1 Target Applications

ISAP was designed with a focus on robustness against implementation attacks including *both* side-channel and fault attacks. In particular, special care has been taken when designing the mode so that implementations of ISAP without any primitive-level countermeasures provide already a higher baseline with respect to protection against side-channel and fault attacks compared to most other authenticated encryption schemes including CCM [22] and GCM [23]. Hence, ISAP is of particular interest for *all* applications where robustness against side-channel and fault attacks is crucial, including various IoT applications, firmware updates of devices, various NFC and smartcard applications, bitstream encryption of FPGAs, etc. In the following, we outline ISAP's protection claims against implementation attacks.

1.1 Plaintext Confidentiality under DPA Attacks

One quite unique feature of ISAP's mode is the fact that it does not enable DPA-based plaintext recovery attacks during authenticated decryption. This is essential in situations such as firmware updates where the plaintexts could carry sensitive information like cryptographic keys. In case of an online/single-pass AEAD scheme, an attacker could query the decryption with a constant nonce and varying ciphertexts, therewith forcing constant key stream blocks that get combined with varying ciphertext blocks. A simple DPA-style attack could then be used to learn the key stream blocks, and thus the corresponding plaintext blocks. Such attacks do not require the extraction of cryptographic keys, but can still be used to undermine the security and integrity of security-critical systems. The two-pass construction of ISAP prevents this type of attack by starting the decryption only after the authenticity of the ciphertext and nonce was successfully verified.

1.2 Differential Power Analysis (DPA)

One of the main design goals of ISAP is inherent protection from certain classes of powerful side-channel attacks that recover the secret key, such as DPA [19]. This is achieved through the usage of the leakage-resilient re-keying function ISAPRK that derives unique session keys for encryption/authentication from the long term key and the nonce. ISAPRK can be viewed as a sponge variant of the classical GGM construction [14]. By limiting the absorption rate during re-keying, one can reduce the number of possible inputs to a permutation call per inner part to 2, which renders classical DPA attacks impractical.

1.3 Differential Fault Analysis (DFA)

DFA [2] attacks exploit the difference between results of repeated executions of cryptographic computations with and without fault injection. During authenticated encryption, fresh nonces ensure that the session keys are unique for each encryption, which prevents DFA attacks.

In the case that the attacker can force multiple queries with the same inputs to Isap (e.g., same ciphertext/nonce/tag during decryption), Isap provides enhanced resilience against the straightforward application of DFA attacks. While Luo et al. [20] show how DFA attacks can be applied to Keccak-based MAC constructions, in the case of Isap, a single fault injection per decryption is not sufficient to learn information about the long-term key. The long term key is only used within IsapRK, which by itself cannot be directly attacked via classical DFA since the attacker never gets to see any output directly. A multi-fault strategy, as outlined in [12], is still possible but requires roughly the quadratic amount of faulted decryptions when compared to the numbers reported in [20]. More importantly, it requires precise combinations of multiple fault injections, both in terms of timing and location, which is considered to be impractical.

1.4 Statistical (Ineffective) Fault Attacks (SFA/SIFA)

SFA [13] and SIFA [6] are fault attack techniques that are, in contrast to DFA, applicable to many AEAD schemes, including online/single-pass variants, and without assumptions such as nonce repetition or release of unverified plaintext. These attacks are especially interesting since it was shown that they are also applicable to masked implementations, whereas SIFA can even work in cases where masking is combined with typical fault countermeasure techniques [6].

Both attacks have in common that they require the attacker to call a certain cryptographic building block (e.g., permutation) with varying inputs. In principle, SFA can be applicable when AEAD schemes perform a final key addition before generating an output [5], which is not the case for Isap. SIFA, on the other hand, can be used in the initialization phase of almost all AEAD schemes, similarly to what was shown for the Keccak-based AEAD schemes Ketje and Keyak [8]. However, in the case of Isap, the 1-bit rate during IsapRK limits the number of inputs per permutation call to 2 and thus severely limits the capabilities of SIFA, which usually requires several hundred calls with varying inputs [8] in practice.

2 Planned Tweak Proposal

We are planning to change the recommendation order of the Isap instances as follows:

1. Isap-A-128a (primary recommendation)
2. Isap-K-128a
3. Isap-A-128
4. Isap-K-128

This change is motivated by (1) the significantly better performance of Ascon-p on 32-bit devices, (2) the noticeably lower area requirements of Ascon-p-based Isap instances in hardware. The specification of the individual Isap instances remains the same.

3 Implementation Aspects

In this section, we outline various implementation aspects of Isap. First, we discuss advantages and performance comparisons of Isap in software. We then present a comparison

of Isap FPGA implementations with variants of the NIST standardized AES GCM mode. Finally, we present performance numbers for Isap that can be achieved on a low-end 32-bit RISC-V microprocessor in combination with a recently proposed compact hardware accelerator for Ascon-p.

3.1 Software

In the following, we compare the performance of Isap to (protected) versions of other schemes. For benchmark numbers, we mainly rely on the third-party analysis done by Guo, Standaert, Wang, and Yu [16] that compares the performance of IsapMac (as used in Isap-K-128a) to AES CBC-MAC implementations utilizing various degrees of masking.

In their analysis a 32-bit ARM core is used to compute a MAC in a side-channel protected manner. More specifically, the authors look at the case of using masked CBC-MACs with 2 to 10 shares. The resulting numbers are then compared to an unprotected version of IsapMac that already provides DPA protection at mode-level.

In the case of very short messages with 16 bytes and 2 share implementations, CBC-MAC is noticeably faster than IsapMac. This advantage diminishes however quickly, either with increasing message length, or with increasing masking order. In the case of 160 byte message length and 4 (resp., 8) share implementations, IsapMac is already about 2.3 (resp., 9.0) times faster than CBC-MAC. Our implementations show that the Ascon-based variants of Isap typically perform even better on these platforms [7].

3.2 FPGA

In the following, we compare the performance of Isap to AES GCM. To allow for an easier comparison, all presented performance metrics are derived from 7-series Xilinx FPGA platforms.

As can be seen in Table 1, area and performance of unprotected AES GCM implementations [17] are roughly on par with Isap, which does offer protection/hardening against side-channel/fault attacks out of the box. However, even if we only consider the overhead of 1st-order Threshold Implementations (TI) for AES GCM, the area increases significantly while the throughput drops. Note that the *fast* version of AES GCM TI does reach very high throughput numbers (15.24 Gbit/s), however only if combined with an RNG (cost not included in the numbers) that can deliver randomness at a rate of up to 175.24 Gbit/s, which is impractical [21].

Table 1: FPGA metrics of Isap compared to the NIST standardized AES GCM mode. The columns SCA and FI indicate if the designs offer some protection against side-channel/fault-injection attacks.

	FPGA	Slices	SCA	FI	Throughput [Mbit/s]	Throughput /Slices
AES GCM [17]	Artix-7	393	✗	✗	700	1.78
AES GCM [17]	Artix-7	781	✗	✗	2 200	2.81
AES GCM TI [21]	Virtex-7	3 422	✓	✗	180	0.05
AES GCM TI (fast) [21]	Virtex-7	38 211	✓	✗	(15 240)	(0.39)
Isap-A-128a [18]	Artix-7	622	✓	✓	1 110	1.78
Isap-K-128a [18]	Artix-7	924	✓	✓	1 560	1.68

3.3 RISC-V Co-Processor

In the following, we present performance numbers for Isap which can be achieved using a 32-bit RISC-V microprocessor in combination with a recently proposed compact hardware accelerator for Ascon-p that requires only 4.7 kGE, or about half the area of dedicated co-processor designs [24].

The accelerator can be used for all permutation-based cryptographic schemes that utilize Ascon's permutation. With Isap and Ascon's family of modes for AEAD and hashing, one can perform AEAD and hashing with a performance of about 2 cycles/byte, or about 4 cycles/byte if protection against fault attacks and power analysis is desired. This roughly corresponds to speed-up factors of about 50 to 80, when compared to corresponding software implementations (cf. Table 2).

When using the compact co-processor with the Isap mode, protection/hardening against implementation attacks such as DPA/DFA/SFA/SIFA is already provided on mode-level. Furthermore, the hardware accelerated computation of Ascon-p also noticeably increases the difficulty of SPA/Template attacks. For a more detailed analysis of the provided implementation security, we refer to Section 5 in [24].

Table 2: Performance metrics of a low-end 32-bit RISC-V microprocessor with/without 1-round hardware acceleration for Ascon-p (`HW-A`)

Implementations	Cycles/Byte			Binary Size (Bytes)
	64 B	1536 B	long	
`ascon128-C (-O3)`	162.0	110.8	106.5	11 716
`ascon128-C (-Os)`	248.5	171.6	168.3	2 104
`ascon128-ASM + HW-A`	4.2	2.2	2.1	888
`isapa128a-ASM + HW-A`	29.1	5.2	4.2	1 844
`isapa128-ASM + HW-A`	73.6	7.7	5.0	2 552

4 New proofs/arguments supporting the security claims

The original Isap submission file already had an explanation of how the black-box security results on the keyed duplex [3] and on the sponge hash function [1] applied to the Isap mode. With respect to leakage resilience, Dobraunig and Mennink considered leakage resilience of the duplex [9] and the suffix sponge [11], and explained how the results combine to leakage resilience of the integral Isap mode [10]. This proof has been expanded and worked out in more precision in [7]. In detail, this article derives an exact security bound on the leakage resilient authenticated encryption security of the generic Isap mode under the assumption that each permutation call leaks a limited amount of data, λ bits, non-adaptively.

Guo, Pereira, Peters, and Standaert [15] independently considered leakage resilience of the Isap mode. They focus on the confidentiality of IsapEnc specifically (authenticity of IsapMac is only sketched). On the other hand, they consider a different leakage assumption, namely that leakages are hard-to-invert, and in this way it complements the security proof of [7]. Degabriele, Janson, and Struck [4] independently considered leakage resilience of sponge based authenticated encryption schemes, also in the bounded leakage model. Their construction strongly resembles Isap with the following notable changes: in encryption the output of IsapRK is used in its entirety to generate the state before keystream generation (so no feedforward of the nonce), and in authentication the output of IsapRK is used as tag, so no final primitive call is made. Finally, the construction is

in fact based on a transformation instead of a permutation. The conference publication contained a significantly better security bound than [7], but it was flawed. The analysis has been fixed in the corresponding ePrint article: both the bound and analysis are now comparable to [7], and they can be seen as a justification of the soundness of the Isap mode.

References

[1] G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche. "On the Indifferentiability of the Sponge Construction". In: *EUROCRYPT 2008*. Ed. by N. P. Smart. Vol. 4965. LNCS. Springer, 2008, pp. 181–197.

[2] E. Biham and A. Shamir. "Differential Fault Analysis of Secret Key Cryptosystems". In: *CRYPTO '97*. Ed. by B. S. Kaliski Jr. Vol. 1294. LNCS. Springer, 1997, pp. 513–525.

[3] J. Daemen, B. Mennink, and G. Van Assche. "Full-State Keyed Duplex with Built-In Multi-user Support". In: *ASIACRYPT 2017*. Ed. by T. Takagi and T. Peyrin. Vol. 10625. LNCS. Springer, 2017, pp. 606–637.

[4] J. P. Degabriele, C. Janson, and P. Struck. "Sponges Resist Leakage: The Case of Authenticated Encryption". In: *ASIACRYPT 2019*. Ed. by S. D. Galbraith and S. Moriai. Vol. 11922. LNCS. Springer, 2019, pp. 209–240.

[5] C. Dobraunig, M. Eichlseder, T. Korak, V. Lomné, and F. Mendel. "Statistical Fault Attacks on Nonce-Based Authenticated Encryption Schemes". In: *ASIACRYPT 2016*. Vol. 10031. LNCS. 2016, pp. 369–395.

[6] C. Dobraunig, M. Eichlseder, T. Korak, S. Mangard, F. Mendel, and R. Primas. "SIFA: Exploiting Ineffective Fault Inductions on Symmetric Cryptography". In: *IACR Trans. Cryptogr. Hardw. Embed. Syst.* 2018.3 (2018), pp. 547–572.

[7] C. Dobraunig, M. Eichlseder, S. Mangard, F. Mendel, B. Mennink, R. Primas, and T. Unterluggauer. "Isap v2.0". In: *IACR Trans. Symmetric Cryptol.* 2020.S1 (2020), pp. 390–416.

[8] C. Dobraunig, S. Mangard, F. Mendel, and R. Primas. "Fault Attacks on Nonce-Based Authenticated Encryption: Application to Keyak and Ketje". In: *SAC 2018*. Vol. 11349. LNCS. Springer, 2018, pp. 257–277.

[9] C. Dobraunig and B. Mennink. "Leakage Resilience of the Duplex Construction". In: *ASIACRYPT 2019*. Ed. by S. D. Galbraith and S. Moriai. Vol. 11923. LNCS. Springer, 2019, pp. 225–255.

[10] C. Dobraunig and B. Mennink. *Leakage Resilience of the ISAP Mode: a Vulgarized Summary*. NIST Lightweight Cryptography Workshop 2019. 2019.

[11] C. Dobraunig and B. Mennink. "Security of the Suffix Keyed Sponge". In: *IACR Trans. Symmetric Cryptol.* 2019.4 (2019), pp. 223–248.

[12] C. Dobraunig, B. Mennink, and R. Primas. *Exploring the Golden Mean Between Leakage and Fault Resilience and Practice*. Cryptology ePrint Archive, Report 2020/200. https://eprint.iacr.org/2020/200. 2020.

[13] T. Fuhr, É. Jaulmes, V. Lomné, and A. Thillard. "Fault Attacks on AES with Faulty Ciphertexts Only". In: *FDTC*. IEEE Computer Society, 2013, pp. 108–118.

[14] O. Goldreich, S. Goldwasser, and S. Micali. "How to construct random functions". In: *J. ACM* 33.4 (1986), pp. 792–807.

[15] C. Guo, O. Pereira, T. Peters, and F.-X. Standaert. "Towards Low-Energy Leakage-Resistant Authenticated Encryption from the Duplex Sponge Construction". In: *IACR Trans. Symmetric Cryptol.* 2020.1 (2020), pp. 6–42.

[16] C. Guo, F.-X. Standaert, W. Wang, and Y. Yu. "Efficient Side-Channel Secure Message Authentication with Better Bounds". In: *IACR Trans. Symmetric Cryptol.* 2019.4 (2019), pp. 23–53.

[17] Helion Technology Limited. "AES-GCM cores". In: (accessed: 09/2020). URL: https://www.heliontech.com/aes_gcm.htm.

[18] ISAP Team. "ISAP Hardware Package". In: (accessed: 09/2020). URL: https://github.com/isap-lwc/isap-hardware-package.

[19] P. C. Kocher, J. Jaffe, and B. Jun. "Differential Power Analysis". In: *CRYPTO '99*. Ed. by M. J. Wiener. Vol. 1666. LNCS. Springer, 1999, pp. 388–397.

[20] P. Luo, Y. Fei, L. Zhang, and A. A. Ding. "Differential Fault Analysis of SHA-3 Under Relaxed Fault Models". In: *J. Hardw. Syst. Secur.* 1.2 (2017), pp. 156–172.

[21] N. Mentens, V. Miskovsky, M. Novotny, and J. Vliegen. *High-speed Side-channel-protected Encryption and Authentication in Hardware*. Cryptology ePrint Archive, Report 2018/1088. https://eprint.iacr.org/2018/1088. 2018.

[22] National Institute of Standards and Technology. *NIST Special Publication 800-38C: Recommendation for Block Cipher Modes of Operation: The CCM Mode for Authentication and Confidentiality*. 2004.

[23] National Institute of Standards and Technology. *NIST Special Publication 800-38D: Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC*. 2007.

[24] S. Steinegger and R. Primas. *A Fast and Compact Accelerator for Ascon and Friends*. Cryptology ePrint Archive, Report 2020/1083. https://eprint.iacr.org/2020/1083. 2020.