A Sub-Set Fault Analysis attack on ASCON

Priyanka Joshi[1] **and Bodhisatwa Mazumdar**[1]

[1]Discipline of Computer Science & Engineering, Indian Institute of Technology, Indore, India.
[1]phd1801201001@iiti.ac.in, bodhisatwa@iiti.ac.in

ABSTRACT

ASCON, designed by Dobraunig et al.[1] is an authenticated encryption, selected as the first choice for a lightweight use case in the CAESAR competition in February 2019. In this work, we investigate vulnerabilities of ASCON against fault analysis. We observe that the use of 128-bit random nonce makes it resistant against many cryptanalysis techniques like differential, linear, etc. and their variants. However, XORing the key just before releasing the tag T (a public value) creates a trivial attack path. Also, the S-Box demonstrates a non-random behavior towards subset cryptanalysis. We observe that if the 3rd bit of the S-box input is set to zero, then XoR of the last two output bits is zero, with a probability of 0.625, i.e., this characteristic is present in 10 out of 16 cases. Our subset fault analysis(SSFA) attack uses this property to retrieve the 128-bit secret key. The SSFA attack can uniquely retrieve the key of full-round ASCON with the complexity of 2^{64}.

1 Objective

ASCON is designed to operate efficiently and securely in highly-constrained environments like the Internet of Things (IoT), where fault attacks make a potent threat. This work aims to evaluate the security of ASCON against a class of fault analysis attacks.

2 ASCON Block Cipher

ASCON is a sponge based cipher with 320-bits state. The initial state of ASCON consists of 64-bit constant IV followed by 128-bit secret key K and Nonce N of 128-bits. The 320-bit sponge state is divided into five 64-bit words x_0, x_1, x_2, x_3, and x_4 as $\{S = S_r||S_c = x_0||x_1||x_2||x_3||x_4\}$. The encryption is partitioned into four stages: initialization, associated-data, plaintext, and finalization. In encryption, it iteratively applies an SPN-based round transformation p which consists of three sub-transformations p_C, p_S and p_L in the same order, $\{p = p_C \circ p_S \circ p_L\}$. The sub-transformation p_C adds a round-constant c_r to the register word x_2 of the state S, $\{x_2 = x_2 \oplus c_r\}$. p_S is a non-linear transformation that represents the substitution layer. The substitution layer consists of 64 parallel instances of a 5-bit S-box $S(x)$. The five inputs of the S-box are taken from five 64-bit register words x_0 to x_4, considering one bit from each word, where x_0 acts as the MSB and x_4 as the LSB of the S-box input. The sub-transformation p_L is a set of linear functions Σ_i that provides diffusion within each register word separately.

3 Threat Model

We assume the attacker is capable of inducing bit-reset fault in a 64-bit word in the input of substitution operation at the last round of the finalization stage in ASCON encryption. The bit-set/reset faults can be induced using laser beam profiling with high precision[2].

4 The Proposed Attack

The SSFA works in two phases:

Phase-I (*Subset fault analysis using key partitioning*) - First, we partition the 128-bit key into n-bit sub-keys (Sk), where n is assumed to be a power of 2. Hence, the total number of subkeys is $N_{sk} = \frac{128}{n}$. Each subkey is a linear combination of n key bits, where coefficients of a linear combination depend on the target S-boxes used for analysis. The subkey Sk_i can be expressed as: $Sk_i = a_i k_i \oplus a_{i+1} k_{i+1} \oplus \cdots \oplus a_{i*n-2} k_{i*n-2} \oplus a_{i*n-1} k_{i*n-1}$. Instead of using key bits directly, we use parity of each subkey (P_{sk}) for our subset analysis, where P_i is one-bit value of Sk_i. Thus, key hypothesis for S-box j is a set $K^{(j)} = \{P_0^{(j)}, P_1^{(j)}, \ldots, P_{N_{sk}-1}^{(j)}\}, P_i^{(j)} \in \{0,1\}, 0 \leq i \leq N_{sk}-1$ So, there are $2^{N_{sk}}$ combinations for each key hypothesis $K^{(j)}$. Consider an example, for $n = 32$, there will be four subkeys. So, for each S-box j, the key hypothesis is a set

$K^{(j)} = \{P_0^{(j)}, P_1^{(j)}, P_2^{(j)}, P_3^{(j)}\}$ with 2^4 possible values for each $K^{(j)}$. The Phase-I estimates the parity of $K^{(j)}$ for each S-box j. It works as depicted in Figure1.

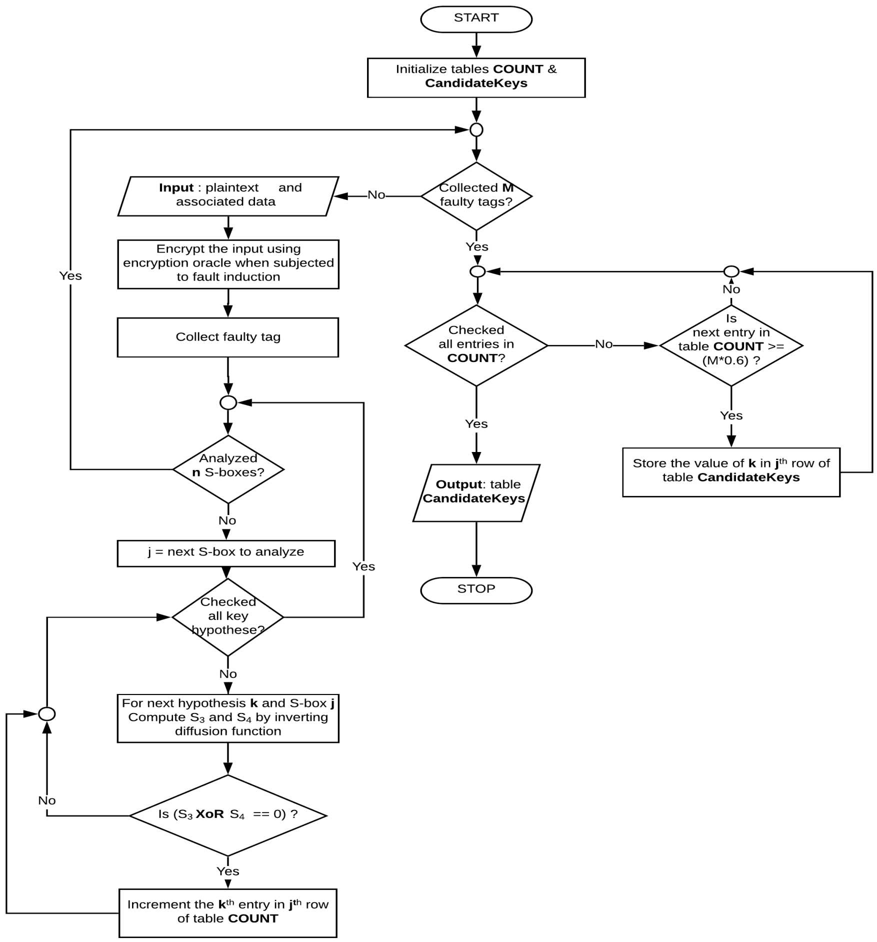

Figure 1. Subset fault analysis using key partitioning

Phase-II (*Key analysis using partition parity*) - Once the correct parity of $K^{(j)}$ is obtained from Phase-I, the combinations that do not satisfy the parity can be eliminated. In the exhaustive search space of $2^{N_{sk}}$ combinations, $2^{N_{sk}/2}$ combinations have even parity, and the other half have odd parity. Thus, half of the key hypotheses are excluded for each S-box. Now, for each of the remaining key hypotheses, formulate N_{Sk} sets of n-linear equations, where each equation in a set corresponds to one of the n S-boxes under analysis. On solving one set of linear equations, we receive n key bits. The complete 128-bit key is then obtained by concatenating N_{Sk} sets of n-key bits. We check for the correctness of the derived key. If the correct key is not determined, we repeat the process on subsequent key hypotheses.

5 Experiments and Results

To verify the proposed attack, we have simulated it on a C implementation of ASCON-128. We performed experiments for $n = 32$, and $n = 64$ with randomly generated plaintext and associated-data pairs while ensuring unique nonce for each encryption. We notice that, in our attack, 70-100 faulty tags can recover the embedded key in the device. Unlike other statistical attacks[3], the number of required faulty encryptions is independent of partition size n because, in our proposed fault model, a single fault in x_2 causes 1-bit faults in all 64 S-boxes. Also, for $n = 32$, Phase-I returns 2^3 candidate key guesses for each S-box, and 32 such S-boxes are required to be analyzed. Hence, it requires $2^{3*32} = 2^{96}$ search operations to retrieve the correct key. Whereas, for $n = 64$, Phase-I returns 2 candidate key guesses for each S-box, and 64 S-boxes are needed to be analyzed. So, it takes 2^{64} search operations to recover the correct key, which is a significant reduction in key search space.

6 Conclusion

We demonstrate that our SSFA attack can recover the entire secret key of full-round ASCON with 70-100 faulty tags and search complexity of 2^{64}. Hence, in the Subset Fault Analysis Model, ASCON-128 does not achieve a 128-bit security level as claimed by designers and fails to attain 112-bit level security, which is the primary requirement for NIST-LWC's consideration.

References

[1] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2. Submission to Round 1 of the NIST Lightweight Cryptography project, 2019.

[2] Cyril Roscian, Alexandre Sarafianos, Jean-Max Dutertre, and Assia Tria. Fault model analysis of laser-induced faults in SRAM memory cells. In *2013 Workshop on Fault Diagnosis and Tolerance in Cryptography, Los Alamitos, CA, USA, August 20, 2013*, pages 89–98, 2013.

[3] Keyvan Ramezanpour, Paul Ampadu, and William Diehl. A statistical fault analysis methodology for the ascon authenticated cipher. In *IEEE International Symposium on Hardware Oriented Security and Trust, HOST 2019, McLean, VA, USA, May 5-10, 2019*, pages 41–50, 2019.