IACR Transactions on Symmetric Cryptology
ISSN 2519-173X, Vol. 0, No. 0, pp. 0–0. DOI:10.46586/tosc.v0.i0.0-0

Diving Deep into the Weak Keys of Round Reduced Ascon

Raghvendra Rohit[1] and Santanu Sarkar[2,3]

[1] Cryptography Research Centre, Technology Innovation Institute, Abu Dhabi, UAE
raghvendra.rohit@tii.ae
[2] Indian Institute of Technology Madras, Chennai, India santanu@iitm.ac.in
[3] Ruhr-University Bochum, Germany

Abstract. At ToSC 2021, Rohit *et al.* presented the first distinguishing and key recovery attacks on 7 rounds Ascon without violating the designer's security claims of nonce-respecting setting and data limit of 2^{64} blocks per key. So far, these are the best attacks on 7 rounds Ascon. However, the distinguishers require (impractical) 2^{60} data while the data complexity of key recovery attacks exactly equals 2^{64}. Whether there are any practical distinguishers and key recovery attacks (with data less than 2^{64}) on 7 rounds Ascon is still an open problem.
In this work, we give positive answers to these questions by providing a comprehensive security analysis of Ascon in the weak key setting. Our first major result is the 7-round cube distinguishers with complexities 2^{46} and 2^{33} which work for 2^{82} and 2^{63} keys, respectively. Notably, we show that such weak keys exist for any choice (out of 64) of 46 and 33 specifically chosen nonce variables. In addition, we improve the data complexities of existing distinguishers for 5, 6 and 7 rounds by a factor of $2^8, 2^{16}$ and 2^{27}, respectively. Our second contribution is a new theoretical framework for weak keys of Ascon which is solely based on the algebraic degree. Based on our construction, we identify $2^{127.99}$, $2^{127.97}$ and $2^{116.34}$ weak keys (out of 2^{128}) for 5, 6 and 7 rounds, respectively. Next, we present two key recovery attacks on 7 rounds with different attack complexities. The best attack can recover the secret key with 2^{63} data, 2^{69} bits of memory and $2^{115.2}$ time. Our attacks are far from threatening the security of full 12 rounds Ascon, but we expect that they provide new insights into Ascon's security.

Keywords: Ascon · Weak keys · Cube attack · Algebraic degree

1 Introduction

Undoubtedly, one of the main security criterion of a keyed cryptographic primitive is its random behavior for any randomly selected key from the entire key space. It is often difficult to guarantee this criterion, as there might exist some keys, often termed as *weak keys*, for which the strength (distinguishability or key recovery) of a primitive may differ significantly. This is evident from a wide range of attacks on symmetric ciphers in the weak key setting [BB93, Haw98, FMS01, KM07, Men17, Kha19, GLR+20, LIMS21].

A typical weak key attack consists of two steps: (1) finding a weak key set and (2) ensuring that the complexity of a distinguisher or key recovery attack is less than the number of weak keys, both of them being challenging tasks. Some promising generic weak key attacks are the invariant and nonlinear invariant subspace attacks [LAAZ11, LMR15, TLS16, Bey18] which have seen applications to block ciphers only.

This work focuses on weak key analysis (from the algebraic degree perspective) of permutation-based authenticated encryption with associated data (AEAD) scheme Ascon, designed by Dobraunig, Eichlseder, Mendel, and Schläffer [DEMS16, DEMS21]. Being one of the winners of the CAESAR competition (the Competition for Authenticated Encryption: Security, Applicability, and Robustness) [CAE] and currently a finalist of the US National Institute of Standards and Technology (NIST) lightweight cryptographic standardization project [Nat19], Ascon has received substantial third-party security evaluation.

The state-of-the-art analyses on Ascon could be divided into two categories: (1) distinguishing attacks on the underlying public permutation and (2) attacks targeting the Ascon AEAD. Examples of the former include differential/linear distinguishers [DEMS15, DEM15, BDKW19], limited-birthday distinguishers [GPT21], zero-sum distinguishing attacks [DEMS15, Tod15, GRW16, YLW⁺19], and subspace trails [LTW18]. The latter category is more relevant to our work as concrete cryptanalysis is performed on Ascon AEAD. Some of the existing results are provable security claims [JLM14], state recovery attacks [DKM⁺17], differential-linear cryptanalysis [DEMS15, LLL21], forgery attacks [DEMS15, LZWW17, GPT21], cube attack and its variants [DEMS15, LDW17, LZWW17, RHSS21].

Among all the aforementioned cryptanalytic results, the best attacks on Ascon in the AEAD context considering the two design requirements ([DEMS16, Chapter 2]), namely (1) nonce value should not be repeated for a fixed key and (2) the data limit per key is 2^{64} blocks, can reach only 7 (out of 12) rounds due to Rohit *et al.* [RHSS21]. However, their distinguishers complexity, i.e., 2^{60} is still not practical while the data complexity of key recovery attacks equals 2^{64}. Furthermore, it is surprising that there is no weak key analysis of Ascon till date. Thus, it is worth investigating the weak key security of Ascon and identifying whether there are any practical distinguishers and key recovery attacks (with data less than 2^{64}) on 7 rounds. Table 1 gives a summary of the attacks on Ascon. We now list our contributions.

Our Contributions. We present a comprehensive security analysis of round-reduced Ascon in the weak key setting without violating any of Ascon's security claims. Our contributions are threefold and summarized as follows.

1. Practical distinguishers for up to 7 rounds: We identify a set of keys and a set of nonce variables (say d out of 64) such that the algebraic degree of the output bits is at most $d-1$ in nonce variables. In particular, for 7 rounds, we find that for any fixed set of $d = 46$ (resp. 33) nonce variables out of $\binom{64}{46}$ (resp. $\binom{64}{33}$) choices, there are 2^{82} (resp. 2^{63}) keys where the algebraic degree of the output bits is at most 45 (resp. 32). This gives distinguishers[1] with complexities 2^{46} and 2^{33}. To the best of our knowledge, these are the first practical distinguishers for 7-round Ascon. Furthermore, in a weak key scenario, our choice of $d = 13$ (9), 24 (17) and 46 (33) improve the complexities of existing distinguishers (which works for all keys) for 5, 6 and 7 rounds by a factor of 2^3 (2^8), 2^8 (2^{16}) and 2^{13} (2^{27}), respectively (see Table 1).

   The source codes of the distinguishers are publicly available at https://github.com/blacksegal/ascon_weak_key_analysis.

2. Theoretical framework of weak keys: We provide the theoretical construction of a weak key space solely based on the algebraic degree. Our central idea is to partition the key space such that for any key in the weak key space, there must exist a set of d nonce variables which achieves an algebraic degree of at most $d-1$ after r rounds. We show that this criterion holds for $2^{127.99}$, $2^{127.97}$ and $2^{116.34}$ keys (out of 2^{128}) with $d = 13$, 24 and 46 for $r = 5, 6$ and 7, respectively. In addition, we find a subset of these keys with $d = 9$, 17 and 33 where the number of keys are $2^{104.1}$, $2^{103.92}$ and $2^{94.67}$ for 5, 6 and 7 rounds, respectively. Moreover, we give structural

[1] The XOR sum of all 2^d values of the output equals zero with probability 1.

Table 1: Summary of attacks on Ascon in the nonce-respecting setting

Key recovery

#Rounds	Data	Time	Method	#Keys	Validity	Source
7/12	$2^{77.2}$	2^{104}	Conditional cube	2^{128}	✗	[LDW17]
7/12	2^{64}	2^{123}	Cube	2^{128}	✓	[RHSS21]
7/12	$2^{77.2}$	2^{77}	Conditional cube	2^{117}	✗	[LDW17]
7/12	$\mathbf{2^{64}}$	$\mathbf{2^{97}}$	**Cube**	$\mathbf{2^{116.34}}$	✓	**Section 6.1**
7/12	$\mathbf{2^{63}}$	$\mathbf{2^{115.2}}$	**Cube**	$\mathbf{2^{116.34}}$	✓	**Section 6.2**

Distinguishers

#Rounds	Data	Time	Method	#Keys	Validity	Source
5/12	2^{17}	2^{17}	Degree	2^{128}	✓	[DEMS15]
5/12	2^{16}	2^{16}	Division Property	2^{128}	✓	[RHSS21]
5/12	$\mathbf{2^{13}}$	$\mathbf{2^{13}}$	**Degree**	$\mathbf{2^{115}}$	✓	**Section 4.2**
5/12	$\mathbf{2^{9}}$	$\mathbf{2^{9}}$	**Degree**	$\mathbf{2^{111}}$	✓	**Section 4.2**
6/12	2^{33}	2^{33}	Degree	2^{128}	✓	[DEMS15]
6/12	2^{31}	2^{31}	Division Property	2^{128}	✓	[RHSS21]
6/12	$\mathbf{2^{24}}$	$\mathbf{2^{24}}$	**Degree**	$\mathbf{2^{104}}$	✓	**Section 4.2**
6/12 †	$\mathbf{2^{18}}$	$\mathbf{2^{18}}$	**Degree**	$\mathbf{2^{110}}$	✓	**Section 4.3**
6/12	$\mathbf{2^{17}}$	$\mathbf{2^{17}}$	**Degree**	$\mathbf{2^{95}}$	✓	**Section 4.2**
7/12	2^{60}	2^{60}	Division Property	2^{128}	✓	[RHSS21]
7/12	$\mathbf{2^{46}}$	$\mathbf{2^{46}}$	**Degree**	$\mathbf{2^{82}}$	✓	**Section 4.2**
7/12	$\mathbf{2^{33}}$	$\mathbf{2^{33}}$	**Degree**	$\mathbf{2^{63}}$	✓	**Section 4.2**

✗: Although a generic attack but violates the required data limit of $\leq 2^{64}$ per key, and hence, is invalid.

†: An experimental distinguisher with a success probability of 0.63.

properties of weak keys such as (1) indices where key bits are equal and/or unequal and (2) Hamming weight of a weak key, which are crucial for key recovery attacks.

3. Key recovery attacks on 7 rounds: We present two key recovery attacks on 7 rounds Ascon with different attack complexities. Our first attack requires 2^{64} data, 2^{70} bits of memory and 2^{97} time[2] while the second attack requires 2^{63} data, 2^{69} bits of memory and $2^{115.2}$ time (see Table 1). Although the time complexity of the latter attack is marginal, it answers the question "*Is there a key recovery attack on 7-round Ascon with less than 2^{64} data ?*" posed by [RHSS21].

Outline of the Paper. The rest of the paper is organized as follows. In Section 2, we define our notation and some well-known relevant cryptanalytic techniques. Section 3 gives the specification of Ascon along with our attack settings. We present the practical weak key distinguishers of round-reduced Ascon in Section 4. In Section 5, we provide the construction of weak key space of Ascon and their structural properties. Section 6 gives the key recovery attacks on 7-round Ascon in the weak key setting. Finally, we conclude in Section 7 with future research directions.

[2] Offline queries to 7-round Ascon permutation.

2 Notation and Preliminaries

Let A and B be two sets. We use $A \cup B$ (resp. $A \cap B$) to denote the set consisting of elements which are in A or B (resp. A and B), while $A \setminus B$ represents the set which contains elements from A but not in B. For a set A, its cardinality is given by $|A|$. Let $\mathbb{F}_2 = \{0, 1\}$ be the finite field with two elements and $\mathbb{F}_2^n$ denotes the n-dimensional vector space over $\mathbb{F}_2$. For $x, y \in \mathbb{F}_2^n$, $x \oplus y$ and $x \| y$ denote the bitwise XOR and concatenation operations, respectively. In addition, we use "+" to denote all kinds of additions (of integers, field elements, and Boolean functions) and the actual meaning should be clear from the context.

Monomial representation and Boolean functions. For a given $u = (u_0, \cdots, u_{n-1}) \in \mathbb{F}_2^n$, we write the monomial x^u in n variables from $x = (x_0, \cdots, x_{n-1})$ as

$$x^u = \prod_{i=0}^{n-1} x_i^{u_i}. \tag{1}$$

Note that $x^u = 1$ if and only if $u_i \leq x_i$ for all $0 \leq i \leq n-1$.

Let $f : \mathbb{F}_2^n \to \mathbb{F}_2$ be a Boolean function whose Algebraic Normal Form (ANF) is defined by $f(x) = \sum_{u \in \mathbb{F}_2^n} a_u x^u$ where $a_u \in \mathbb{F}_2$. For any $u \in \mathbb{F}_2^n$, we denote its Hamming weight by $\mathbf{wt}(u)$. The algebraic degree of a Boolean function f, represented by $deg(f)$, is defined as $deg(f) = \max\{\mathbf{wt}(u) \mid a_u \neq 0\}$.

Keyed Boolean functions. Let $v = (v_0, \cdots, v_{m-1})$ be m public variables and $k = (k_0, \cdots, k_{n-1})$ be n secret variables. Then, in the context of symmetric ciphers, each output bit can be regarded as a Boolean function $f : \mathbb{F}_2^m \times \mathbb{F}_2^n \to \mathbb{F}_2$ given by

$$f(v, k) = \sum_{u \in \mathbb{F}_2^m} \sum_{w \in \mathbb{F}_2^n} a_{u,w} v^u k^w, \tag{2}$$

where $a_{u,w} \in \mathbb{F}_2$. In Equation 2, $deg(f) = \max\{\mathbf{wt}(u) + \mathbf{wt}(w) \mid a_{u,w} \neq 0\}$. For a fixed key k, which is usually treated as a secret constant in cryptanalysis, we are interested in the algebraic degree in public variables only. Thus, in our work, we focus on $deg(f) = \max\{\mathbf{wt}(u) \mid a_{u,w} \neq 0\}$.

Cube attacks. The cube attack proposed in [Vie07, DS09] analyzes a keyed Boolean function as a black-box polynomial which is tweakable in public variables. Given n secret variables $k = (k_0, \cdots, k_{n-1})$, m public variables $v = (v_0, \cdots, v_{m-1})$, a set of indices $\mathcal{I} = \{i_0, \cdots, i_{d-1}\} \subseteq \{0, \cdots, m-1\}$ and $\bar{\mathcal{I}} = \{0, \cdots, m-1\} \setminus \mathcal{I}$, Equation 2 can alternatively be viewed as

$$f(v, k) = \Big(\prod_{i \in \mathcal{I}} v_i \Big) \cdot t(\bar{I}, k) + q(v, k), \tag{3}$$

where each monomial in the Boolean function q misses at least one variable from $v[\mathcal{I}] = \{v_i \mid i \in \mathcal{I}\}$. Following the terminology of cube attacks, we denote $\mathcal{I}$, $v[\mathcal{I}]$ and a Boolean function $t(\cdot)$ as the *cube indices* set, *cube variables* set, and the *superpoly* of cube monomial $\prod_{i \in \mathcal{I}} v_i$, respectively.

Let $\mathcal{C}_{v[\mathcal{I}]}$ denote the set consisting of all 2^d possible values of the variables in $\mathcal{I}$ while the variables in $\bar{\mathcal{I}}$ are fixed to some constant. We call $\mathcal{C}_{v[\mathcal{I}]}$ as the *d-dimensional cube*, and summing $f(v, k)$ over it (also termed as the *cube-sum*) gives the superpoly $t(\bar{I}, k)$. More precisely, we have

$$\bigoplus_{\mathcal{C}_{v[\mathcal{I}]}} f(v, k) = t(\bar{I}, k). \tag{4}$$

Finding the ANF of a superpoly or showing that a certain cube monomial does not appear in the ANF are the essence of cube attack and its variants [ADMS09, KMN10, DS11, HWX+17]. The former is typically exploited for key recovery attacks while the latter is used as a distinguisher. There are division property [Tod15, TM16] based automated techniques which can recover the ANF of a superpoly [TIHM17, WHT+18, WHG+19, HLM+20, HLLT20, HSWW20]. However, in this work, we concentrate on distinguishers and show how they can be utilized for key recovery attacks in case of Ascon without the need of any automated tools.

3 Specification of Ascon and Attack Settings

Ascon [DEMS16, DEMS21], designed by Dobraunig *et al.*, is a permutation-based family of authenticated encryption with associated data algorithms (AEAD). The Ascon AEAD algorithm takes as inputs a secret key K, a nonce N, a block header AD (a.k.a associated data) and a message M. It then outputs a ciphertext C of the same length as M, and an authentication tag T which authenticates the associated data AD and the message M. It operates in a sponge-duplex mode [BDPA11, Dae12] (as shown in Figure 1)[3] using the iterative permutations p^a and p^b with a and b rounds, respectively. Ascon has two variants, namely Ascon-128 and Ascon-128a. Table 2 lists these two variants along with their recommended parameters.

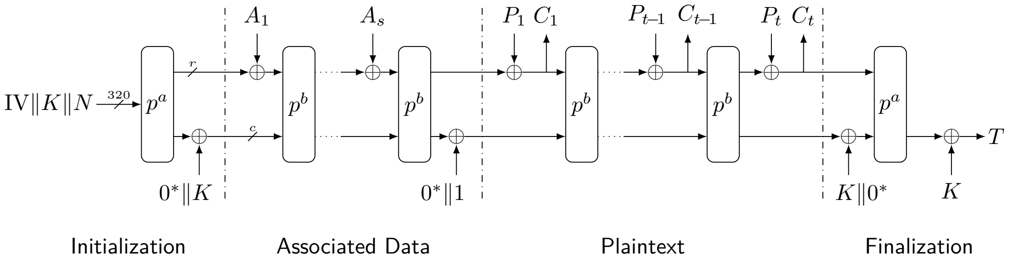

Figure 1: Ascon's mode of operation (encryption phase)

Table 2: Ascon variants and their recommended parameters

Name	State size	Rate r	Size of			Rounds		
			Key	Nonce	Tag	p^a	p^b	IV
Ascon-128	320	64	128	128	128	12	6	80400c0600000000
Ascon-128a	320	128	128	128	128	12	8	80800c0800000000

3.1 The Ascon Permutation

The core permutation p of Ascon is based on a substitution permutation network (SPN) based design paradigm. It operates on a 320-bit state arranged into five 64-bit words and is defined as $p : p_L \circ p_S \circ p_C$. The state at the input of the r-th round is denoted by $X_0^r \| X_1^r \| X_2^r \| X_3^r \| X_4^r$ while $Y_0^r \| Y_1^r \| Y_2^r \| Y_3^r \| Y_4^r$ represents the state after the p_S layer. We use $X_i^r[j]$ (resp. $Y_i^r[j]$) to denote the j-th bit (starting from left) of X_i^r (resp. Y_i^r). We now describe the three steps p_C, p_S, and p_L in detail (superscripts are removed for simplicity).

[3] Thanks to TikZ for Cryptographers [Jea16].

Addition of constants (p_C). As shown in Figure 2, an 8-bit constant is added to the bits $56, \cdots, 63$ of word X_2 at each round.

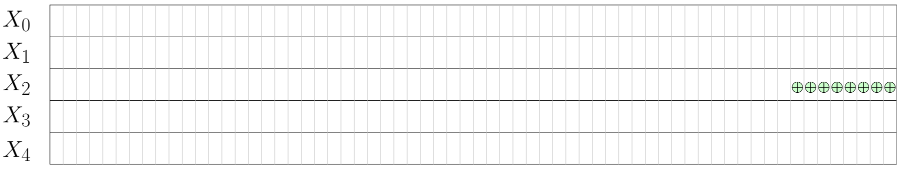

Figure 2: Addition of constants (p_C)

Substitution layer (p_S). A 5-bit Sbox is applied on each of the 64 columns (see Figure 3). Let $(x_0, x_1, x_2, x_3, x_4)$ and $(y_0, y_1, y_2, y_3, y_4)$ denote the input and output of the Sbox, respectively. Then the algebraic normal form (ANF) of the Sbox is given in Equation 5. Note that here x_i and y_i are the bits of the word X_i and Y_i, respectively.

$$\begin{cases} y_0 = x_4x_1 + x_3 + x_2x_1 + x_2 + x_1x_0 + x_1 + x_0 \\ y_1 = x_4 + x_3x_2 + x_3x_1 + x_3 + x_2x_1 + x_2 + x_1 + x_0 \\ y_2 = x_4x_3 + x_4 + x_2 + x_1 + 1 \\ y_3 = x_4x_0 + x_4 + x_3x_0 + x_3 + x_2 + x_1 + x_0 \\ y_4 = x_4x_1 + x_4 + x_3 + x_1x_0 + x_1 \end{cases} \tag{5}$$

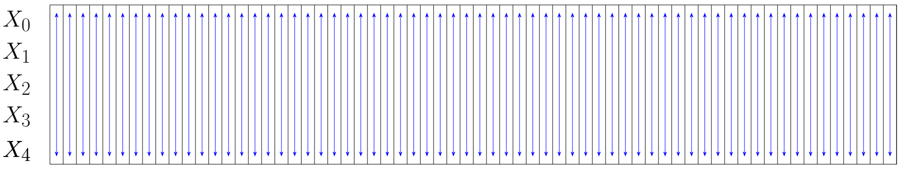

Figure 3: Substitution layer p_S

Linear diffusion layer (p_L). Each 64-bit word is updated by a linear operation Σ_i which is defined in Equation 6 and also illustrated in Figure 4. Here $\ggg$ is the right cyclic shift operation over a 64-bit word.

$$\begin{cases} X_0 \leftarrow \Sigma_0(Y_0) = Y_0 + (Y_0 \ggg 19) + (Y_0 \ggg 28) \\ X_1 \leftarrow \Sigma_1(Y_1) = Y_1 + (Y_1 \ggg 61) + (Y_1 \ggg 39) \\ X_2 \leftarrow \Sigma_2(Y_2) = Y_2 + (Y_2 \ggg 1) + (Y_2 \ggg 6) \\ X_3 \leftarrow \Sigma_3(Y_3) = Y_3 + (Y_3 \ggg 10) + (Y_3 \ggg 17) \\ X_4 \leftarrow \Sigma_4(Y_4) = Y_4 + (Y_4 \ggg 7) + (Y_4 \ggg 41) \end{cases} \tag{6}$$

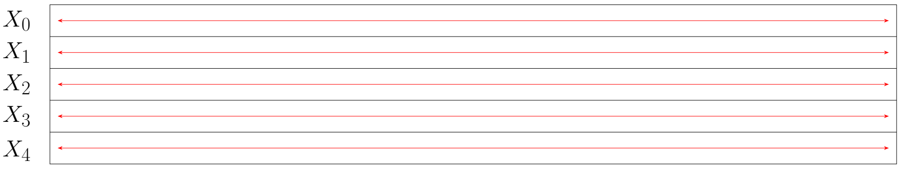

Figure 4: Linear diffusion layer p_L

3.2 Attack Configuration and Targets

We focus on the initialization phase of Ascon (see Figure 5) reduced to $r \in \{5, 6, 7\}$ out of 12 rounds, in the nonce-respecting setting. In our attacks, we query the Ascon oracle q times for distinct nonces N_i and the known-plaintexts P_i, and obtain the corresponding ciphertext blocks C_i for $i = 0, \cdots, q-1$. For a fixed key K and $AD = \phi$, we denote these queries by $C_i \leftarrow \mathsf{Ascon}(K, N_i, \phi, M_i)$ where the tag is omitted. We consider two attacks as follows.

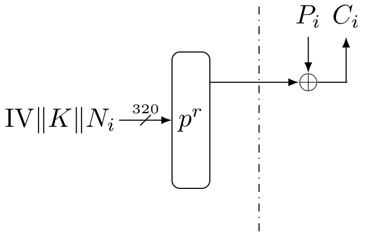

Figure 5: Our attack configuration

Distinguishing attacks. Our goal is to find a set of keys denoted by WK^r and a set of cube indices $\mathcal{I} = \{i_0, i_1, \cdots, i_{d-1}\}$ such that

$$\bigoplus_{\mathcal{C}_{\boldsymbol{v}[\mathcal{I}]}} Y_0^r[i] = 0, \quad \text{for all } 0 \leq i \leq 63.$$

Moreover, we aim to achieve small values of d to have low data complexity.

Key recovery attacks. What are the complexities of recovering $K \in \mathsf{WK}^r$? Is there a key recovery attack with data less than 2^{64} ?

In the following, we only give the distinguishing and key recovery attacks on Ascon-128 in detail. However, they are equally applicable to Ascon-128a as the underlying permutation is the same for both variants.

4 Practical Weak Key Distinguishers

In this section, we present the distinguishers for round-reduced Ascon with practical data complexities, in the weak key setting. We explain the idea of constructing the distinguishers and give concrete examples.

4.1 Core Idea of Distinguishers

Our main idea is to reduce the algebraic degree of the output bits (in terms of nonce bits $v_0, \cdots, v_{127}$) by imposing certain conditions on nonce bits $v_0, \cdots, v_{127}$ and the secret key bits $k_0, \cdots, k_{127}$. We achieve this in two steps as follows.

Step 1. Constraints on nonce bits [RHSS21]. The idea is similar to the one proposed in [RHSS21]. We first look at the Sbox output after round 1 as given in Equation 7

$$\begin{cases} Y_0^0[i] = v_{i+64}k_i + v_i + (rc_i + k_{i+64})k_i + rc_i + k_{i+64} + k_i \cdot \mathrm{IV}[i] + k_i + \mathrm{IV}[i] \\ Y_1^0[i] = v_{i+64} + v_i(rc_i + k_{i+64}) + v_i k_i + v_i + (rc_i + k_{i+64})k_i + rc_i + k_{i+64} + k_i + \mathrm{IV}[i] \\ Y_2^0[i] = v_{i+64}v_i + v_{i+64} + rc_i + k_{i+64} + k_i + 1 \\ Y_3^0[i] = v_{i+64} \cdot \mathrm{IV}[i] + v_{i+64} + v_i \cdot \mathrm{IV}[i] + v_i + rc_i + k_{i+64} + k_i + \mathrm{IV}[i] \\ Y_4^0[i] = v_{i+64}k_i + v_{i+64} + v_i + k_i \cdot \mathrm{IV}[i] + k_i, \end{cases} \tag{7}$$

where rc_i is a round constant bit, and $rc_i = 1$, for $i \in \{56, 57, 58, 59\}$ and zero otherwise. Setting $v_i = v_{i+64}$ reduces Equation 7 to Equation 8 as follows.

$$\begin{cases} Y_0^0[i] = v_i k_i + v_i + (rc_i + k_{i+64})k_i + rc_i + k_{i+64} + k_i \cdot \mathrm{IV}[i] + k_i + \mathrm{IV}[i] \\ Y_1^0[i] = v_i(rc_i + k_{i+64}) + v_i k_i + (rc_i + k_{i+64})k_i + rc_i + k_{i+64} + k_i + \mathrm{IV}[i] \\ Y_2^0[i] = rc_i + k_{i+64} + k_i + 1 \\ Y_3^0[i] = rc_i + k_{i+64} + k_i + \mathrm{IV}[i] \\ Y_4^0[i] = v_i k_i + k_i \cdot \mathrm{IV}[i] + k_i \end{cases} \tag{8}$$

The condition $v_i = v_{i+64}$ in Equation 7 ensures that $Y_2^0[i]$ and $Y_3^0[i]$ in Equation 8 are independent of the nonce variable v_i for a fixed i.

Step 2. Constraints on key bits. We now make $Y_1^0[i]$ independent of v_i in Equation 8 by adding the following constraints on the key bits.

$$\begin{cases} k_i = 1 + k_{i+64}, & i \in \{56, 57, 58, 59\} \\ k_i = k_{i+64}, & i \in \{0, \cdots, 63\} \setminus \{56, 57, 58, 59\} \end{cases} \tag{9}$$

Upper bounds on the algebraic degree. Combining Equation 8 and Equation 9, the algebraic degrees of words 0, 1, 2, 3 and 4 after the Sbox and linear layer of round 1 are 1, 0, 0, 0 and 1, respectively. Accordingly, the upper bounds on the algebraic degree (in variables $v_0, \cdots, v_{63}$) for up to 7 rounds can be easily computed by hand and are given in Table 3.

Table 3: Upper bounds on the algebraic degree of Ascon in cube variables

Round r	Bits in word				
	X_0^r	X_1^r	X_2^r	X_3^r	X_4^r
1	**1**	0	0	0	1
2	**1**	1	1	**2**	1
3	**2**	3	3	3	2
4	**6**	6	5	5	5
5	**12**	11	10	11	12
6	**23**	22	23	24	23
7	**45**	47	47	47	46

4.2 Weak Key Distinguishers (Theoretical)

In this section, we give two explicit examples of practical weak key distinguishers based on our degree observations in Table 3. For the sake of brevity, we focus on 7-round Ascon. Here we present a 46 (resp. 33) dimensional cube. We show that there are 2^{82} (resp. 2^{63}) keys for which the cube sum after 7 rounds is always zero for these cubes.

Example 1. Let $\mathcal{I}_1 = \{0, 1, \cdots, 45\}$ and define

$$\mathsf{WK}^7_{\mathcal{I}_1} := \{(k_0, \cdots, k_{127}) \mid k_i = k_{i+64} \text{ for } i \in \mathcal{I}_1\}.$$

Then $|\mathsf{WK}^7_{\mathcal{I}_1}| = 2^{46} \cdot 2^{(64-46)\cdot 2} = 2^{82}$. Now, for the 46-dimensional cube satisfying $v_i = v_{i+64}$ for $i \in \mathcal{I}_1$, the cube sum after 7 rounds is always zero for all $K \in \mathsf{WK}^7_{\mathcal{I}_1}$. This holds as the algebraic degree in cube variables after 7 rounds is at most 45 (see Table 3).

Example 2. Let $\mathcal{I}_2 = \{0, 1, \cdots, 32\}$ and define

$$\begin{aligned}\mathsf{WK}^7_{\mathcal{I}_2} :=& \{(k_0, \cdots, k_{127}) \mid k_i = k_{i+64} = 0 \text{ for } i \in \mathcal{I}_2\} \bigcup \\ & \{(k_0, \cdots, k_{127}) \mid k_i = k_{i+64} = 1 \text{ for } i \in \mathcal{I}_2\}.\end{aligned} \tag{10}$$

Then $|\mathsf{WK}^7_{\mathcal{I}_2}| = 2^{(64-33)\cdot 2} \times 2 = 2^{63}$. Now, for the 33-dimensional cube satisfying $v_i = v_{i+64}$ for $i \in \mathcal{I}_2$, the cube sum after 7 rounds is always zero for all $K \in \mathsf{WK}^7_{\mathcal{I}_2}$. To see why it holds, note that the quadratic term exists only in X_3^2 (see row 2 in Table 3). These quadratic terms do not appear in the ANF of X_3^2 if $k_i = k_{i+64} = 0$ or $k_i = k_{i+64} = 1$ for all $i \in \mathcal{I}_2$. Since the algebraic degree is 1 after 2 rounds, the maximum degree can be $2^5 = 32$ after 7 rounds. Note that $\mathsf{WK}^7_{\mathcal{I}_2} \subset \mathsf{WK}^7_{\mathcal{I}_1}$.

We follow a similar approach for 5 and 6 rounds. Some examples along with the number of weak keys are depicted in Table 4.

Table 4: Examples of weak keys

Round r	$\mathcal{I}_1$	$\mathsf{WK}^r_{\mathcal{I}_1}$	$\mathcal{I}_2$	$\mathsf{WK}^r_{\mathcal{I}_2}$
5	$\{0, \cdots, 12\}$	2^{115}	$\{0, \cdots, 8\}$	2^{111}
6	$\{0, \cdots, 23\}$	2^{104}	$\{0, \cdots, 16\}$	2^{95}
7	$\{0, \cdots, 45\}$	2^{82}	$\{0, \cdots, 32\}$	2^{63}

Experimental verification. We have experimentally verified all the weak key distinguishers till 7 rounds. The source codes are available at https://github.com/blacksegal/ascon_weak_key_analysis.

Remark 1. In the above discussion and in Table 4, we have only considered the keys corresponding to one specific indices set. However, there exist multiple indices sets, and consequently, the number of weak keys is the union of keys corresponding to these sets. In Section 5, we define the weak key space for the key recovery attacks.

4.3 Weak Key Distinguishers (Experimental)

In this section, we give some distinguishers for 6 rounds based on our experimental observations. We present some small size cubes which give distinguisher with good success probability in the weak key setting.

We take different cube indices $\mathcal{I} \subset \{0, \cdots, 63\}$ with

$$\begin{cases} k_i = 1 + k_{i+64}, & i \in \{56, 57, 58, 59\} \cap \mathcal{I} \\ k_i = k_{i+64}, & i \in \mathcal{I} \setminus \{56, 57, 58, 59\}. \end{cases} \tag{11}$$

Experimentally we first calculate the probability $p < 1/2$ of a superpoly to be nonzero. After the cube sum, we have a vector $a = (a_0, \cdots, a_{63}) \in \mathbb{F}_2^{64}$. For a random source, we have $\Pr(a_i = 1) = \frac{1}{2}$, for $0 \le i \le 63$. On the other hand, in the case of Ascon, we have $\Pr(a_i = 1) < \frac{1}{2}$. So, we give a threshold $\mathcal{T}$ and if $x = |\{i \in \{0, \cdots, 63\} \mid a_i = 1\}| < \mathcal{T}$, we can assume the source is Ascon; otherwise we assume the source is random. We present this idea in Algorithm 1.

Our distinguisher fails in two ways as follows:

Algorithm 1: Experimental distinguisher setup

```
Input: Threshold $\mathcal{T}$
Output: Ascon or random source
if the number of nonzero coordinates $x$ after the cube sum satisfies $x \leq \mathcal{T}$, then
    Output Ascon
else
    Output random source
end
```

1. The source is Ascon but $x > \mathcal{T}$. This happens with probability $\sum_{i=\mathcal{T}+1}^{64} \binom{64}{i} p^i (1-p)^{64-i}$.

2. The source is random but $x \leq \mathcal{T}$. This happens with probability $\sum_{i=0}^{\mathcal{T}} \binom{64}{i} \frac{1}{2^{64}}$.

Accordingly, the success probability of our distinguisher is given by

$$1 - 0.5\Big(\sum_{i=\mathcal{T}+1}^{64} \binom{64}{i} p^i (1-p)^{64-i} + \sum_{i=0}^{\mathcal{T}} \binom{64}{i} \frac{1}{2^{64}} \Big). \tag{12}$$

Experimental results. We did the experiments for 2^{16} random keys with a random cube each time. Since each key gives 64 superpolies, we have 2^{22} superpolies in total. We take the average over these values. Some of the distinguishers are listed below.

- $|\mathcal{I}| = 23$: In this case $p = 0.22$ after 6 rounds. The threshold $\mathcal{T} = 22$ gives the best distinguisher with success probability 0.99.

- $|\mathcal{I}| = 22$: In this case $p = 0.42$ after 6 rounds. The threshold $\mathcal{T} = 29$ gives the best distinguisher with success probability 0.74.

- $|\mathcal{I}| = 21$: In this case $p = 0.48$ after 6 rounds. The threshold $\mathcal{T} = 31$ gives the best distinguisher with success probability 0.56.

Next we did the experiments with a fixed cube and tried to identify some cubes for 6 rounds which give better distinguishers than a random cube. The results are shown in Table 5.

Table 5: Examples of weak key distinguishers (experimental) for 6 rounds Ascon

$\lvert\mathcal{I}\rvert$	$\mathcal{I}$	p	$\mathcal{T}$	Sucess prob.
21	{2, 3, 7, 8, 9, 10, 11, 13, 16, 17, 18, 19, 27, 28, 32, 33, 34, 36, 37, 42, 63}	0.29	25	0.96
20	{2, 3, 7, 8, 9, 10, 11, 13, 16, 17, 18, 19, 27, 28, 32, 33, 34, 36, 37, 42}	0.35	27	0.89
19	{2, 3, 7, 8, 9, 10, 11, 13, 16, 17, 18, 19, 27, 28, 33, 34, 36, 37, 42}	0.41	29	0.77
18	{2, 3, 7, 8, 9, 10, 13, 16, 17, 18, 19, 27, 28, 33, 34, 36, 37, 42}	0.46	30	0.63

5 The Weak Key Space of Ascon

Finding a weak key set for a cipher is typically a challenging task unless some specific structural properties exist within the cipher. In this section, we show how to construct a weak key space of Ascon based on the algebraic degree. We first explain the idea of weak keys, define them formally and then present the theoretical construction which works for $r \geq 2$ rounds of Ascon. Next, we identify some additional structural properties of weak keys which are crucial for key recovery attacks. Finally, we present a combinatorial method to count the number of such weak keys (lower bounds) for $r = 5, 6$ and 7 rounds.

5.1 Defining Weak Key Space

Our main idea is to partition the key space such that for any key in the weak key space, there exists a d-dimensional cube for which all 64 superpolies are zero after r rounds, for some d. We denote this set of keys by $\mathsf{WK}^r[d]$ and formally define it in Definition 1.

Definition 1. [Weak key space $\mathsf{WK}^r[d]$]. Let $d \geq 1$ and $K \in \mathsf{WK}^r[d]$. Then there exists a d-dimensional cube such that the algebraic degree of $Y_0^r[i]$ in d cube variables is at most $d-1$ after r rounds, for all $0 \leq i \leq 63$.

Remark 2. Definition 1 only is about the existence of a cube. Finding such a cube and then using it in a key recovery attack is discussed later in Section 6.

Construction of weak keys set. Let $r \geq 2$ be the number of rounds. We construct $\mathsf{WK}^r[d]$ (specific to r-round Ascon only) based on the key constraints given in Equation 9. Let $\mathcal{I} = \{i_0, \cdots, i_{d-1}\} \subseteq \{0, \cdots, 63\}$. Define

$$\mathsf{WK}^r_{\mathcal{I}}[d] \coloneqq \{(k_0, \cdots, k_{127}) \mid (\star\star) \text{ holds}\}, \tag{13}$$

where $(\star\star)$ is given by

$$(\star\star) \coloneqq \begin{cases} k_i = 1 + k_{i+64}, & i \in \{56, 57, 58, 59\} \cap \mathcal{I} \\ k_i = k_{i+64}, & i \in \mathcal{I} \setminus \{56, 57, 58, 59\}. \end{cases} \tag{14}$$

Note that for different choices of $\mathcal{I}$, the keys might be repeated. Thus, in order to make the counting of weak keys easy to follow, we redefine Equation 14 by introducing a middle condition, as given in Equation 15.

$$\left\{\begin{array}{ll} k_i = k_{i+64}, & i \in \mathcal{I} \text{ and } i \neq 56, 57, 58, 59 \\ k_i = 1 + k_{i+64}, & i \notin \mathcal{I} \text{ and } i \neq 56, 57, 58, 59 \\ k_i = 1 + k_{i+64}, & i \in \mathcal{I} \text{ and } i \in \{56, 57, 58, 59\} \end{array}\right\} \text{ for } 0 \leq i \leq 63. \tag{15}$$

The first two conditions in Equation 15 considers the sets $\mathcal{I}$ which do not contain $\{56, 57, 58, 59\}$ as a subset. On the other hand, the last condition (along with the first two) considers the sets $\mathcal{I}$ which contain at least one of the indices 56, 57, 58, 59. Since there are $\binom{64}{d}$ many choices of $\mathcal{I}$, the set $\mathsf{WK}^r[d]$ is given by

$$\mathsf{WK}^r[d] = \bigcup_{\mathcal{I}=\{i_0,\cdots,i_{d-1}\}\subseteq\{0,\cdots,63\}} \mathsf{WK}^r_{\mathcal{I}}. \tag{16}$$

We now give the values of d for 5, 6 and 7 rounds Ascon.

Specifying d for Ascon. We set $d = 13, 24$ and 46 for $r = 5, 6$ and 7 rounds, respectively. Definition 1 holds for these choices as there exist d cube variables of the form $v_{i_0} = v_{i_0+64}, \cdots, v_{i_{d-1}} = v_{i_{d-1}+64}$. For this setting, the algebraic degrees are 12, 23 and 45 after 5, 6 and 7 rounds, respectively (see Table 3).

5.2 Structural Properties of Weak Keys

We state four properties (relevant for key recovery attacks) of the weak key space based on its construction and the values of the cube sum.

5.2.1 Weak Keys and Equality among Key Indices

In Property 1, we give the relationship between a weak key and the number of indices where key bits are equal and/or unequal.

Property 1. Let $K \in \mathsf{WK}^r[d]$ and $\mathcal{I} = \{i_0, \cdots, i_{d-1}\} \subseteq \{0, \cdots, 63\} \setminus \{56, 57, 58, 59\}$. Let $\boldsymbol{v}[\mathcal{I}] = \{v_i \mid \text{for } \ i \in \mathcal{I}\}$ be a d-dimensional cube and set $v_i = v_{i+64}$ for all $i \in \mathcal{I}$. Then the following assertions hold.

1. There exists at least $d - 4$ indices in $\mathcal{I}$ where $k_i = k_{i+64}$.

2. If $\bigoplus_{\mathcal{C}_{\boldsymbol{v}[\mathcal{I}]}} Y_0^r[j] \neq 0$ for all $0 \leq j \leq 63$, then there exist at least 1 index in $\mathcal{I}$ such that $k_i = 1 + k_{i+64}$.

3. If $\bigoplus_{\mathcal{C}_{\boldsymbol{v}[\mathcal{I}]}} Y_0^r[j] \neq 0$ for all $0 \leq j \leq 63$ and for all $\mathcal{I}$, then there exist at least $60 - d - 1$ indices in $\{0, \cdots, 63\} \setminus \{56, 57, 58, 59\}$ such that $k_i = 1 + k_{i+64}$.

Remark 3. The last two assertions in Property 1 hold for any K in the 2^{128} key space.

5.2.2 Smaller Subset of $\mathsf{WK}^r[d]$

We find a subset of $\mathsf{WK}^r[d]$ for which there exists $2^{r-2} + 1$ cube variables (instead of d) which can reach an algebraic degree of at most 2^{r-2} after r rounds. This subset is formally given in Property 2.

Property 2. Let $r \geq 2$ and $\mathsf{WK}^r[d]$ be given. Then there exists $\mathsf{sWK}^r[d'] \subset \mathsf{WK}^r[d]$ with $d' = 2^{r-2} + 1 < d$.

Proof. Let $\mathcal{I}' = \{i_0, \cdots, i_{d'-1}\} \subset \{0, \cdots, 63\}$. Define

$$\mathsf{sWK}^r_{\mathcal{I}'}[d'] \coloneqq \{(k_0, \cdots, k_{127}) \mid (\star\star_0) \text{ or } (\star\star_1) \text{ holds}\}, \tag{17}$$

where $(\star\star_0)$ and $(\star\star_1)$ are given by

$$(\star\star_0) \coloneqq \left.\begin{cases} k_i = k_{i+64} = 0, & i \in \mathcal{I}' \text{ and } i \neq 56, 57, 58, 59 \\ k_i = 1 + k_{i+64}, & i \notin \mathcal{I}' \text{ and } i \neq 56, 57, 58, 59 \\ k_i = 0, \ k_{i+64} = 1, & i \in \mathcal{I}' \text{ and } i \in \{56, 57, 58, 59\} \end{cases}\right\} \text{ for } 0 \leq i \leq 63, \tag{18}$$

$$(\star\star_1) \coloneqq \left.\begin{cases} k_i = k_{i+64} = 1, & i \in \mathcal{I}' \text{ and } i \neq 56, 57, 58, 59 \\ k_i = 1 + k_{i+64}, & i \notin \mathcal{I}' \text{ and } i \neq 56, 57, 58, 59 \\ k_i = 1, \ k_{i+64} = 0, & i \in \mathcal{I}' \text{ and } i \in \{56, 57, 58, 59\} \end{cases}\right\} \text{ for } 0 \leq i \leq 63. \tag{19}$$

For any $K \in \mathsf{sWK}^r[d']$, $K \in \mathsf{WK}^r[d]$ as well, meaning that $\mathsf{sWK}^r[d'] \subset \mathsf{WK}^r[d]$. To see why $d' = 2^{r-2} + 1$ holds, note that if we set $v_i = v_{i+64}$ for all $i \in \mathcal{I}'$, then X_3^2 becomes linear in cube variables (see Table 3). Thus, the algebraic degree in cube variables is at most 1 after 2 rounds and at most 2^{r-2} after $2 + r$ rounds. □

5.2.3 Hamming Weight of a Weak Key

For any $K \in \mathsf{WK}^r[d]$, we identify some relations on the Hamming weight of K by simply observing the cube sum. More precisely, we give bounds on the Hamming weight of a weak key. The bounds are given in Property 3 where the core observation is based on the definition of $\mathsf{sWK}^r[2^{r-2}]$ in Property 2.

Property 3. Let $r \geq 2$, $K \in \mathsf{WK}^r[d]$ and $\mathcal{I} = \{i_0, \cdots, i_{2^{r-2}}\} \subset \{0, \cdots, 63\} \backslash \{56, 57, 58, 59\}$. Let $\boldsymbol{v}[\mathcal{I}] = \{v_i \mid \text{for } \ i \in \mathcal{I}\}$ be a d-dimensional cube and set $v_i = v_{i+64}$ for all $i \in \mathcal{I}$. If $\bigoplus_{\mathcal{C}_{\boldsymbol{v}[\mathcal{I}]}} Y_0^r[j] \neq 0$ for all $0 \leq j \leq 63$, then $1 \leq \mathbf{wt}(k_{i_0}, \cdots, k_{i_{2^{r-2}}}) \leq 2^{r-2}$.

5.2.4 Relationship between $\mathsf{WK}^r[d]$ and $\mathsf{WK}^r[d+1]$

From the construction of weak keys, it is trivial to see that a key which is weak under $\mathsf{WK}^r[d+1]$ is also a weak key under $\mathsf{WK}^r[d]$. In particular, for any $K \in (\mathsf{WK}^r[d] \bigcup \mathsf{WK}^r[d+1])$, there exists d cube variables whose cube sum is zero after r rounds. The above discussion is summarized in Property 4.

Property 4. Let $\mathsf{WK}^r[d]$ and $\mathsf{WK}^r[d+1]$ be given. Then for any $K \in (\mathsf{WK}^r[d] \bigcup \mathsf{WK}^r[d+1])$, there exists a d-dimensional cube such that $\bigoplus_{\mathcal{C}_{\boldsymbol{v}[\mathcal{I}]}} Y_0^r[j] = 0$ for all $0 \leq j \leq 63$.

Weak key sets for Ascon. We use Property 4 and define the weak key set WK^r for r-round Ascon as

$$\mathsf{WK}^r := \bigcup_{i=d}^{64} \mathsf{WK}^r[i]. \tag{20}$$

Similarly, we have $\mathsf{sWK}^r \subset \mathsf{WK}^r$ which can be constructed based on Property 2.

5.3 Dimension of Weak Keys

In this section, we present a combinatorial method to count the number of weak keys, i.e. $|\mathsf{WK}^r|$ and $|\mathsf{sWK}^r|$ for a given r.

5.3.1 Size of WK^r

We first focus on Equation 15 and Equation 20 in detail for a given d. Let $\mathcal{I} = \{i_0, \cdots, i_{d-1}\}$ be a set of d indices selected out of $\{0, \cdots, 63\}$. We count the keys for all choices of $\mathcal{I}$ and without repetition. The following cases are possible based on the construction in Equation 15.

- **Case 1:** $\mathcal{I}$ does not contain $\{56, 57, 58, 59\}$. In this case, the key bits corresponding to $\{56, 57, 58, 59\}$ can take all 2^8 values. The d indices in $\mathcal{I}$ satisfy $k_i = k_{i+64}$ while the remaining $60 - d$ indices satisfy $k_i = 1 + k_{i+64}$. Accordingly, for all choices of $\mathcal{I}$, the number of keys is $\binom{60}{d} \cdot 2^d \cdot 2^{60-d} \cdot 256$.

- **Case 2:** $\mathcal{I}$ contains at least one index from $\{56, 57, 58, 59\}$. The $d - 1$ indices in $\mathcal{I}$ satisfy $k_i = k_{i+64}$ while the remaining $60 - d + 1$ indices satisfy $k_i = 1 + k_{i+64}$. The number of keys is $\binom{60}{d-1} \cdot 2^{d-1} \cdot 2^{60-d+1} \cdot \alpha_1$, for some constant α_1 (its actual value and reasoning provided later on).

- **Case 3:** $\mathcal{I}$ contains at least two indices from $\{56, 57, 58, 59\}$. The $d - 2$ indices in $\mathcal{I}$ satisfy $k_i = k_{i+64}$ while the remaining $60 - d + 2$ indices satisfy $k_i = 1 + k_{i+64}$. The number of keys is $\binom{60}{d-2} \cdot 2^{d-2} \cdot 2^{60-d+2} \cdot \alpha_2$, for some constant α_2.

- **Case 4:** $\mathcal{I}$ contains at least three indices from $\{56, 57, 58, 59\}$. The $d-3$ indices in $\mathcal{I}$ satisfy $k_i = k_{i+64}$ while the remaining $60-d+3$ indices satisfy $k_i = 1 + k_{i+64}$. The number of keys is $\binom{60}{d-3} \cdot 2^{d-3} \cdot 2^{60-d+3} \cdot \alpha_3$, for some constant α_3.

- **Case 5:** $\mathcal{I}$ contains $\{56, 57, 58, 59\}$. The $d-4$ indices in $\mathcal{I}$ satisfy $k_i = k_{i+64}$ while the remaining $60-d+4$ indices satisfy $k_i = 1 + k_{i+64}$. The number of keys is $\binom{60}{d-4} \cdot 2^{d-4} \cdot 2^{60-d+4} \cdot \alpha_4$, for some constant α_4.

The constant terms α_i's are computed as follows. Let k_{56}, k_{57}, k_{58}, k_{59}, k_{120}, k_{121}, k_{122}, k_{123} be an 8-bit subkey. Then, out of 256 possible values, we compute the number of distinct subkeys where there are at least 1, 2, 3 and 4 indices in $\{56, 57, 58, 59\}$ satisfying $k_i = 1 + k_{i+64}$ for α_1, α_2, α_3 and α_4, respectively. We computed them using a simple Python code (also provided in the Supplementary Material) and obtained $\alpha_1 = 240$, $\alpha_2 = 176$, $\alpha_3 = 80$ and $\alpha_4 = 16$.

Adding the keys in cases (1) – (5) (since their intersection is empty), we have

$$|\mathsf{WK}^r[d]| = 2^{60} \cdot \Bigg(\binom{60}{d} \cdot 256 + \binom{60}{d-1} \cdot 240 + \binom{60}{d-2} \cdot 176 + \binom{60}{d-3} \cdot 80 + \binom{60}{d-4} \cdot 16\Bigg). \tag{21}$$

Since $\mathsf{WK}^r = \bigcup_{i=d}^{64} \mathsf{WK}^r[i]$, we enumerate all keys following an approach similar to $\mathsf{WK}^r[d]$. However, there may be repetitions in $\mathsf{WK}^r[i]$, $\mathsf{WK}^r[i+1], \cdots, \mathsf{WK}^r[64]$. For example, $\mathsf{WK}^r[64] \subset \mathsf{WK}^r[60]$. To avoid ambiguity, we give a lower bound on the size of WK^r by only considering the sets whose intersections are empty. Note that the exact value will not differ significantly from the lower bound for $r = 7$ as $d = 46$ contributes to the majority of keys.[4] At the end, we have

$$|\mathsf{WK}^r| \geq |\mathsf{WK}^r[d]| + \sum_{i=d+1}^{60} \binom{60}{i} \cdot 2^{60} \cdot 2^8. \tag{22}$$

5.3.2 Size of sWK^r

We first find the number of keys that satisfy Equation 18. In particular for a given d', we find the size of $\mathsf{sWK}^r[d']$. Let $\mathcal{I} = \{i_0, \cdots, i_{d-1}\}$ be a set of d indices selected out of $\{0, \cdots, 63\}$. We count the keys for all choices of $\mathcal{I}$ and without repetition. Again, we have 5 cases.

- **Case 1:** $\mathcal{I}$ does not contain $\{56, 57, 58, 59\}$. In this case, the key bits corresponding to $\{56, 57, 58, 59\}$ can take all 2^8 values. The d' indices in $\mathcal{I}$ satisfy $k_i = k_{i+64} = 0$ while the remaining $60 - d'$ indices satisfy $k_i = 1 + k_{i+64}$. Accordingly, the number of keys is $\binom{60}{d'} \cdot 2^{60-d'} \cdot 256$.

- **Case 2:** $\mathcal{I}$ contains at least one index from $\{56, 57, 58, 59\}$. The number of keys is $\binom{60}{d'-1} \cdot 2^{60-d'+1} \cdot \beta_1$, for some constant β_1.

- **Case 3:** $\mathcal{I}$ contains at least two indices from $\{56, 57, 58, 59\}$. The number of keys is $\binom{60}{d'-2} \cdot 2^{60-d'+2} \cdot \beta_2$, for some constant β_2.

- **Case 4:** $\mathcal{I}$ contains at least three indices from $\{56, 57, 58, 59\}$. The number of keys is $\binom{60}{d'-3} \cdot 2^{60-d'+3} \cdot \beta_3$, for some constant β_3.

[4] In fact $|\mathsf{WK}^7| \approx 2^{117}$ if we add the contributions for all $d = 46, \cdots, 64$.

- **Case 5:** $\mathcal{I}$ contains $\{56, 57, 58, 59\}$. The number of keys is $\binom{60}{d'-4} \cdot 2^{60-d'+4} \cdot \beta_4$, for some constant β_4.

The constant terms β_i's are computed as follows. For an 8-bit subkey (k_{56}, k_{57}, k_{58}, k_{59}, k_{120}, k_{121}, k_{122}, k_{123}), we compute the number of distinct subkeys where there are at least 1, 2, 3 and 4 indices in $\{56, 57, 58, 59\}$ satisfying $k_i = 0$ and $k_{i+64} = 1$ for β_1, β_2, β_3 and β_4, respectively. Using the same Python code we obtained $\beta_1 = 175$, $\beta_2 = 67$, $\beta_3 = 13$ and $\beta_4 = 1$.

Adding the keys in cases $(1) - (5)$ (since their intersection is empty), the number of keys that satisfy Equation 18 for a fixed d' is

$$\binom{60}{d'} \cdot 2^{60-d'} \cdot 256 + \binom{60}{d'-1} \cdot 2^{60-d'+1} \cdot 175 + + \binom{60}{d'-2} \cdot 2^{60-d'+2} \cdot 67 \\ + \binom{60}{d'-3} \cdot 2^{60-d'+3} \cdot 13 + \binom{60}{d'-4} \cdot 2^{60-d'+4}. \tag{23}$$

Now, since the intersection of keys corresponding to Equation 18 and Equation 19 is empty, we can multiply Equation 23 by 2 to get the number of keys satisfying Equation 18 and Equation 19. Thus, the sizes of $\mathsf{sWK}^r[d']$ and sWK^r are given as follows.

$$\begin{aligned} |\mathsf{sWK}^r[d']| = 2 \cdot \Bigg(& \binom{60}{d'} \cdot 2^{60-d'} \cdot 256 + \binom{60}{d'-1} \cdot 2^{60-d'+1} \cdot 175 \\ & + \binom{60}{d'-2} \cdot 2^{60-d'+2} \cdot 67 + \binom{60}{d'-3} \cdot 2^{60-d'+3} \cdot 13 \\ & + \binom{60}{d'-4} \cdot 2^{60-d'+4} \Bigg) \end{aligned} \tag{24}$$

$$|\mathsf{sWK}^r| \geq |\mathsf{sWK}^r[d']| + \sum_{i=d'+1}^{60} 2 \cdot \binom{60}{i} 2^{60-i} \cdot 2^8 \tag{25}$$

5.3.3 Lower Bounds and Experimental Verification

Using Equation 22 and Equation 25, we compute the lower bounds on the sizes of WK^r and sWK^r. In Table 6, we list these numbers for $5 - 7$ rounds.

Table 6: Total number of weak keys (lower bounds)

Round r	d	$\|\mathsf{WK}^r\|$	Probability	d'	$\|\mathsf{sWK}^r\|$	Probability
5	13	$2^{127.99}$	$2^{-0.01}$	9	$2^{104.09}$	$2^{-23.91}$
6	24	$2^{127.97}$	$2^{-0.03}$	17	$2^{103.92}$	$2^{-24.08}$
7	46	$2^{116.34}$	$2^{-11.66}$	33	$2^{94.67}$	$2^{-33.33}$

Experimental verification. To verify the correctness of Equation 21 and Equation 24, we computed the number of weak keys for the small parameters: (1) key size equals 16 bits, (2) $d = d' = 4$, and (3) we select the 4 indices $\{1, 2, 3, 4\}$ which are similar to $\{56, 57, 58, 59\}$. Our experimental results matches exactly with these equations. The source codes for verification are publicly available at https://github.com/blacksegal/ascon_weak_key_analysis.

6 Key Recovery Attacks in the Weak Key Setting

In this section, we present key recovery attacks on 7-round Ascon in the weak key setting. We give two attacks: (1) an attack with data complexity 2^{64} and (2) an improved key recovery attack with data complexity 2^{63}.

6.1 Key Recovery Attack with 2^{64} Data

Let $K \in \mathsf{WK}^7$ as defined in Equation 20. Since $|\mathsf{WK}^7| \approx 2^{116.34}$ (see Table 6), the goal is to recover K with complexities (memory and time) strictly less than $2^{116.34}$ Ascon queries. We recover K in two phases, namely (1) *Data collection phase* and (2) *Key recovery phase.* We now explain each phase in detail and discuss the respective attack complexities.

6.1.1 Data Collection Phase

In this phase, we query the Ascon oracle for 2^{64} distinct nonces, empty associated data and 1-block of message. The nonces are chosen as discussed in Section 4.1 and for simplicity, we assume that the message block equals 0^{64} (64 bit zero string) in each of the queries. The ciphertext block is then stored in a hash table $\mathbb{T}$, indexed by the 64-bit value of the nonce. The entire phase is illustrated in Algorithm 2.

```
Algorithm 2: Data collection phase
   Input: Empty hash table $\mathbb{T}$
   Output: Hash table $\mathbb{T}$
 1 Set $M = 0^{64}$                                ▷ 64 bit message block with all zeros
 2 for $i = 0$ to $2^{64} - 1$ do
 3     Set $N = i \| i$                                        ▷ Nonce condition
 4     $C \leftarrow \mathsf{Ascon}(K, N, \phi, M)$                   ▷ Tag is omitted
 5     $\mathbb{T}[i] \leftarrow C$
 6 end
 7 return $\mathbb{T}$
```

Complexity evaluation. The time and memory complexities of this phase are 2^{64} Ascon queries (Line 4 in Algorithm 2) and $2^{64} \cdot 64 = 2^{70}$ bits of memory (Line 5 in Algorithm 2), respectively.

6.1.2 Key Recovery Phase

In this phase, we recover the secret key K. Since $K \in \mathsf{WK}^7$, by Definition 1, there exists a 46-dimensional cube which gives the XOR sum of ciphertexts as 0^{64} after 7 rounds. Further, the keys corresponding to these 46-dimensional cubes should satisfy Equation 15. In our attack, we only need to identify a single set of cube variables (by checking all $\binom{64}{46}$ cubes) and its respective keys by doing local operations on table $\mathbb{T}$. Next, the obtained set of keys is filtered by doing an exhaustive search. An algorithmic description of this phase is provided in Algorithm 3.

Complexity evaluation. The worst case complexities of this phase are $\binom{64}{46} \cdot 2^{46} \approx 2^{96.67}$ memory access to $\mathbb{T}$ and the same number of 64-bit XOR operations in order to recover a 46-dimensional cube (Lines 3-11 in Algorithm 3). Once such a set $\mathcal{I}$ is recovered, then the number of keys corresponding to it satisfying Equation 15 are given as follows.

1. $2^{46+14} \cdot 2^8 = 2^{68}$, if $\mathcal{I}$ does not contain the indices 56, 57, 58 and 59.

Algorithm 3: Key recovery phase

```
   Input: Hash table $\mathbb{T}$
   Output: Secret key $K$
 1 $\mathcal{K} = \{\}$                                          ▷ Empty list of keys
 2 for each $\mathcal{I} = \{i_0, \cdots, i_{45}\} \subset \{0, \cdots, 63\}$ do
 3     $L \leftarrow 0$
 4     for $i = 0$ to $2^{46} - 1$ do
 5         $V_0 \leftarrow 0$
 6         for $j = 0$ to $45$ do
 7             $V_0[i_j] = (i \gg j)\&1$                         ▷ $j$-th bit of integer $i$
 8         end
 9         $L \leftarrow L \oplus \mathbb{T}[V_0]$
10     end
11     if $L == 0$ then
12         for each $K'$ satisfying Equation 15 do
13             $C \leftarrow \mathsf{Ascon}(K', 0^{64}, \phi, 0^{64})$    ▷ Offline computation
14             if $C == \mathbb{T}[0]$ then
15                 Add $K'$ to $\mathcal{K}$
16             end
17         end
18     end
19     Break for loop
20 end
21 Perform exhaustive search on $\mathcal{K}$ to get $K$
22 return $K$
```

2. $2^{45+15} \cdot 2 \cdot 2^6 = 2^{67}$, if $\mathcal{I}$ contains only one of the index 56, 57, 58 and 59.

3. $2^{44+16} \cdot 2^2 \cdot 2^4 = 2^{66}$, if $\mathcal{I}$ contains exactly two of the indices 56, 57, 58 and 59.

4. $2^{43+17} \cdot 2^3 \cdot 2^2 = 2^{65}$, if $\mathcal{I}$ contains exactly three of the indices 56, 57, 58 and 59.

5. $2^{42+18} \cdot 2^4 = 2^{64}$, if $\mathcal{I}$ contains the indices 56, 57, 58 and 59.

These keys are then filtered exhaustively to obtain a set of possible key candidates $\mathcal{K}$ (Lines 11-20 in Algorithm 3). The time complexity of this step is 2^{68} (assuming the worst case) offline Ascon evaluations.[5] Since we are doing a match on 64 bits in Line 14, the size of $\mathcal{K}$ is $2^{68-64} = 2^4$ on average. Finally, we do an exhaustive search on $\mathcal{K}$ (Line 21 in Algorithm 3) to recover K.

Overall, the time complexity is $2^{96.67}$ (memory access) + $2^{96.67}$ (64-bit XORs) + $2^{68}+2^4$ (offline Ascon evaluations). In the worst case, the time complexity is dominated by 2^{97} Ascon evaluations.

Combining the complexities of data collection and key recovery phases, the attack complexities are 2^{64} data, 2^{70} memory (in bits), and 2^{97} time (Ascon evaluations).

6.1.3 Discussion on the units of Time Complexity

The overall time complexity should be in terms of number of Ascon queries (online + offline). Since we use memory accesses to table $\mathbb{T}$ and 64-bit XOR operations, we define

[5] Here, one Ascon evaluation is equivalent to 7-round Ascon permutation computation.

the following scale factors.

$$1 \text{ memory access} \approx 1\ \mathsf{Ascon} \text{ evaluation}$$
$$1\ 64\text{-bit XOR} \approx 2^{-7.2}\ \mathsf{Ascon} \text{ evaluations}$$

The latter is due to the fact that Ascon's bit-sliced implementation requires $21 \times 7 = 147$ 64-bit XORs for 7 rounds.

6.1.4 Discussion on the Recovered Indices Set

We argue that the indices set $\mathcal{I}$ recovered in the data collection phase is correct with very high probability. Let's assume that it is incorrect. In that case, there exists at least one $i \in \mathcal{I}$ for which the key conditions in Equation 15 does not hold. For simplicity, we further assume that there is only one such i. This implies that the cube variable v_i is present in $Y_1^0[i]$, and consequently, in $X_1^1[i]$, $X_1^1[i+3]$ and $X_1^1[i+25]$. Thus, for an incorrect $\mathcal{I}$, the degree increases quickly compared to the correct one. The differences in the degree upper bounds are shown in Table 7. We see that the degree bounds are 45 and 59 for the right and wrong $\mathcal{I}$, respectively, after 7 rounds. Since the dimension of cube is 46, the probability that the wrong $\mathcal{I}$ gives the cube sum as zero in all the 64 output bits is 2^{-64}.

Table 7: Upper bounds on the algebraic degree of Ascon in cube variables. For an incorrect $\mathcal{I}$, the values are shown in $[\cdot]$ and taken from [RHSS21, Section 7.2]

Round r	Bits in word				
	X_0^r	X_1^r	X_2^r	X_3^r	X_4^r
1	1 [1]	0 [1]	0 [0]	0 [0]	1 [1]
2	1 [2]	1 [1]	1 [1]	2 [2]	1 [2]
3	2 [3]	3 [3]	3 [4]	3 [4]	2 [3]
4	6 [7]	6 [8]	5 [7]	5 [7]	5 [6]
5	12 [15]	11 [15]	10 [13]	11 [14]	12 [15]
6	23 [30]	22 [29]	23 [29]	24 [30]	23 [30]
7	45 [59]	47 [59]	47 [60]	47 [60]	46 [58]

Now, for a randomly chosen $K \in \mathsf{WK}^7$, we find the expected number of cubes for which all 64 superpolies are zero after 7 rounds. We have $\mathcal{N} = \binom{64}{46}$ cubes in total. If a cube $\mathcal{C}$ is such that all 64 superpolies are zero after 7 rounds, we say that $\mathcal{C}$ satisfies property $\mathcal{P}$. Now in the weak key setting, there is at least one cube with $\mathcal{P}$. Let us assume that any other cube apart from this cube satisfies $\mathcal{P}$ with probability 2^{-64}. We want to find the number of cubes X which satisfy $\mathcal{P}$. Define a binary random variable X_i which takes 1 if and only if the i-th cube satisfy $\mathcal{P}$. It is clear that $E(X_i) = 2^{-64}$. Thus

$$E(X) = 1 + E(\sum_{i=1}^{\mathcal{N}-1} X_i) = 1 + \sum_{i=1}^{\mathcal{N}-1} E(X_i) = 1 + (\mathcal{N}-1)2^{-64}$$

$$\implies E(X) = 1 + \Big(\binom{64}{46} - 1\Big)2^{-64} \approx 1 + 2^{-12.3}.$$

6.2 Key Recovery Attack with 2^{63} Data

Let $K \in \mathsf{WK}^7$ as defined in Equation 20. We show how to recover K with 2^{63} data and time $< 2^{116.34}$. The attack is again divided into a data collection phase and a key recovery phase, which are described as follows.

6.2.1 Data Collection Phase

We set the cube variables as $v_0, \cdots, v_{58}, v_{60}, \cdots, v_{63}$ and $v_{59} = 0$. We then query the Ascon oracle for 2^{63} distinct nonces, empty associated data and a zero message block similar to Subsubsection 6.1.2. The ciphertext block is then stored in a hash table $\mathbb{T}_1$, indexed by the 64-bit value of the nonce.

Complexity evaluation. The time and memory complexities of this phase are 2^{63} Ascon queries and $2^{63} \cdot 64 = 2^{69}$ bits of memory, respectively.

6.2.2 Key Recovery Phase

We divide this phase into 5 steps. Each of these steps are sequential, i.e., we only move to the next step if specific conditions (mentioned later) are not met. We denote the time complexity of step i by T_i. Further, let $\mathcal{L} = \{56, 57, 58, 59\}$ and $\mathcal{J} = \{0, \cdots, 63\} \setminus \mathcal{L}$.

Step 1: Early filtering with 46 dimension cubes. We apply Algorithm 3 to $\mathcal{J}$ using table $\mathbb{T}_1$. If we find a 46-dimensional cube whose cube sum is zero, then we are done. Else if none of the $\binom{60}{46}$ cube sums are zero, then we proceed to the next step. The time complexity $\mathsf{T}_1 = \binom{60}{46} \cdot 2^{46}$ (memory access) $+ \binom{60}{46} \cdot 2^{46}$ (64-bit XORs).

Since none of the cube sum equals zero, by Property 1, there exist at least 15 and at most 18 indices in $\mathcal{J}$ where $k_i \neq k_{i+64}$. We consider the case for 15, 16, 17 and 18 (not equal) indices in step 2, 3, 4 and 5, respectively.

Step 2: 15 not equal indices. This means there are 45 indices in $\mathcal{J}$ where $k_i = k_{i+64}$ and 15 indices where $k_i \neq k_{i+64}$. Now, to satisfy the definition of $\mathsf{WK}^7[46]$, there must exist at least one index in $\mathcal{L}$ satisfying Equation 15. We select 46-dimensional cubes by choosing 45 variables from $\mathcal{J}$ and 1 variable from $\{56, 57, 58\}$, and then apply Algorithm 3. If none of the cube sums is zero, then $k_{56} = k_{120}$, $k_{57} = k_{121}$, $k_{58} = k_{122}$ and $k_{59} = 1 + k_{123}$. The number of key candidates are $\binom{60}{45} \cdot 2^{60} \cdot 2^4$.

The time complexity of this step is given by

$$\mathsf{T}_2 = \underbrace{\binom{60}{45} \cdot \binom{3}{1} \cdot 2^{46}}_{\text{memory access}} + \underbrace{\binom{60}{45} \cdot \binom{3}{1} \cdot 2^{46}}_{\text{64-bit XORs}} + \underbrace{\binom{60}{45} \cdot 2^{60} \cdot 2^4}_{\text{exhaustive search}}.$$

Step 3: 16 not equal indices. Here, we have 44 indices in $\mathcal{J}$ where $k_i = k_{i+64}$ and 16 indices where $k_i \neq k_{i+64}$. Again, to satisfy the definition of $\mathsf{WK}^7[46]$, there must be at least two indices in $\mathcal{L}$ satisfying Equation 15. We select 46-dimensional cubes by choosing 44 variables from $\mathcal{J}$ and 2 variables from $\{56, 57, 58\}$, and then apply Algorithm 3. If none of the cube sums is zero, then $k_{59} = 1 + k_{123}$ and there exists exactly one i in $\{55, 56, 57\}$ such that $k_i = 1 + k_{i+64}$. The number of key candidates are $\binom{60}{44} \cdot 2^{60} \cdot 48$.

The time complexity is given by

$$\mathsf{T}_3 = \underbrace{\binom{60}{44} \cdot \binom{3}{2} \cdot 2^{46}}_{\text{memory access}} + \underbrace{\binom{60}{44} \cdot \binom{3}{2} \cdot 2^{46}}_{\text{64-bit XORs}} + \underbrace{\binom{60}{44} \cdot 2^{60} \cdot 48}_{\text{exhaustive search}}.$$

Step 4: 17 not equal indices. We have 43 indices in $\mathcal{J}$ where $k_i = k_{i+64}$ and 17 indices where $k_i \neq k_{i+64}$. To satisfy the weak keys definition, there must be at least three indices in $\mathcal{L}$ satisfying Equation 15. We select 46-dimensional cubes by choosing 43 variables from $\mathcal{J}$ and 3 variables from $\{56, 57, 58\}$, and then apply Algorithm 3. If none of the cube sums is zero, then $k_{59} = 1 + k_{123}$ and there exists exactly two i in $\{55, 56, 57\}$ such that $k_i = 1 + k_{i+64}$. The number of key candidates are $\binom{60}{43} \cdot 2^{60} \cdot 48$.

The time complexity is given by

$$\mathsf{T}_4 = \underbrace{\binom{60}{43} \cdot 2^{46}}_{\text{memory access}} + \underbrace{\binom{60}{44} \cdot 2^{46}}_{\text{64-bit XORs}} + \underbrace{\binom{60}{43} \cdot 2^{60} \cdot 48}_{\text{exhaustive search}}.$$

Step 5: 18 not equal indices. Since v_{59} is not a cube variable, we do an exhaustive search on $\binom{60}{42} \cdot 2^{60} \cdot 2^4$ keys. Thus, $\mathsf{T}_5 = \binom{60}{42} \cdot 2^{60} \cdot 2^4$.

Complexity evaluation. Combining steps (1) to (5), the time complexity is given as follows.

$$\begin{aligned}\text{Memory access} &= 2^{46} \cdot \left(\binom{60}{46} + \binom{60}{45} \cdot 3 + \binom{60}{44} \cdot 3 + \binom{60}{43}\right) \approx 2^{95.86}\\ \text{64-bit XORs} &= 2^{46} \cdot \left(\binom{60}{46} + \binom{60}{45} \cdot 3 + \binom{60}{44} \cdot 3 + \binom{60}{43}\right) \approx 2^{95.86}\\ \text{Exhaustive search} &= 2^{68} + 2^{60} \cdot \left(\binom{60}{45} \cdot 16 + \binom{60}{44} \cdot 48 + \binom{60}{43} \cdot 48 + \binom{60}{42} \cdot 16\right)\\ &\approx 2^{115.2}\end{aligned}$$

In summary, the attack requires 2^{63} data, 2^{69} memory (in bits) and $2^{115.2}$ offline Ascon evaluations in the worst case.

6.3 Discussion on Key Recovery Attacks

Here we compare the complexities of the two key recovery attacks discussed in Subsection 6.1 and Subsection 6.2 with that of exhaustive search in the weak key setting. We also briefly give some insights on the possibility of extension of these attacks.

Comparison with exhaustive search. Since the number of weak keys is $2^{116.34}$, the exhaustive search requires $2^{116.34}$ time complexity. For our first attack with 2^{64} data, the time complexity is 2^{97}, and thus, there is an improvement of more than 19 bits over a key space of size $2^{116.34}$.

For our second attack with 2^{63} data, the time complexity is $2^{115.2}$. Although the time complexity is marginally better (around 1 bit) than the exhaustive search, the presented attack is the first key recovery attack on 7-round Ascon with at most 2^{63} data.

We also note that all these complexities are computed in the worst case, i.e., when we go over all steps of the attack in the key recovery phase.

Extending weak key attacks. It is natural to ask whether it is possible to extend our weak key attacks to attacks covering the full key space. Moreover, is it possible to improve the time complexities of existing cube-based attacks [LDW17, RHSS21] on 7 rounds Ascon using the weak key distinguishers. At the moment, Property 1 and Property 3 can certainly reduce the 128-bit key space, but our initial findings suggest that the reduction factor is very low (not even 1 bit). An initial approach in this direction could be to use all smaller sub-cubes of a larger cube to find multiple relations among the key bits. For instance, consider the 6-round Ascon. A 33-dimensional cube gives only 1 relation in key bits if we use the approach of [LDW17]. However, by using Property 1 and Property 3, we could use smaller sub-cubes of dimension 24 and obtain multiple relations among the key bits.

The above mentioned questions certainly need further investigation and therefore, we mention them as an interesting research problem in Section 7.

7 Conclusion

In this work, we have presented the first in-depth weak key security analysis of round-reduced Ascon. We identified two practical distinguishers for 7 rounds with data complexities 2^{46} and 2^{33}, and further improved the state-of-the-art distinguishers complexities by a factor of 2^{8}, 2^{16} and 2^{27} for 5, 6 and 7 rounds, respectively. Moreover, we have shown the existence and construction of a large class of weak keys by simply using algebraic degree arguments. The lower bounds on the number of weak keys are $2^{127.99}$, $2^{127.97}$ and $2^{116.34}$ for 5, 6 and 7 rounds, respectively. We then discussed two key recovery attacks on 7 rounds in the weak key setting with complexities: (1) 2^{64} data, 2^{70} bits of memory and 2^{97} time, and (2) 2^{63} data, 2^{69} bits of memory and $2^{115.2}$ time. Our second attack is the best till now considering the data limit of less than 2^{64} blocks.

Although all our results are in the weak key setting, we believe they will provide new insights to the community in further understanding the security of Ascon. We now list some problems which are worth investigating.

Problem 1. How to extend our weak key attacks to attacks covering the full key space? Is it possible to improve the time complexities of existing cube-based attacks [LDW17, RHSS21] on 7 rounds Ascon using the weak key distinguishers?

Problem 2. We believe that the number of weak keys could be increased by relaxing the success probability of a distinguisher from 1 to some α satisfying $0.5 < \alpha < 1$. This needs further investigation and a starting point could be the presented experimental distinguishers in Section 4.3.

Problem 3. Is there a weak key distinguisher for 8 rounds Ascon?

8 Acknowledgement

The authors would like to thank the reviewers of ToSC 2021 and Léo Perrin for providing us with insightful comments to improve the quality of the paper. Santanu Sarkar acknowledges the Experienced Researchers Fellowship from Alexander von Humboldt Foundation.

References

[ADMS09] Jean-Philippe Aumasson, Itai Dinur, Willi Meier, and Adi Shamir. Cube Testers and Key Recovery Attacks on Reduced-Round MD6 and Trivium. In Orr Dunkelman, editor, *Fast Software Encryption, 16th International Workshop, FSE 2009, Leuven, Belgium, February 22-25, 2009, Revised Selected Papers*, volume 5665 of *Lecture Notes in Computer Science*, pages 1–22. Springer, 2009.

[BB93] Ishai Ben-Aroya and Eli Biham. Differential Cryptanalysis of Lucifer. In Douglas R. Stinson, editor, *Advances in Cryptology - CRYPTO '93, 13th Annual International Cryptology Conference, Santa Barbara, California, USA, August 22-26, 1993, Proceedings*, volume 773 of *Lecture Notes in Computer Science*, pages 187–199. Springer, 1993.

[BDKW19] Achiya Bar-On, Orr Dunkelman, Nathan Keller, and Ariel Weizman. DLCT: A new tool for differential-linear cryptanalysis. In *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part I*, pages 313–342, 2019.

[BDPA11] Guido Bertoni, Joan Daemen, Michaël Peeters, and Gilles Van Assche. Duplexing the Sponge: Single-Pass Authenticated Encryption and Other Applications. In Ali Miri and Serge Vaudenay, editors, *Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*, volume 7118 of *Lecture Notes in Computer Science*, pages 320–337. Springer, 2011.

[Bey18] Tim Beyne. Block Cipher Invariants as Eigenvectors of Correlation Matrices. In Thomas Peyrin and Steven D. Galbraith, editors, *Advances in Cryptology - ASIACRYPT 2018 - 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2-6, 2018, Proceedings, Part I*, volume 11272 of *Lecture Notes in Computer Science*, pages 3–31. Springer, 2018.

[CAE] CAESAR: Call for Submission. http://competitions.cr.yp.to/.

[Dae12] Joan Daemen. Permutation-based Encryption, Authentication and Authenticated Encryption. DIAC 2012, 2012.

[DEM15] Christoph Dobraunig, Maria Eichlseder, and Florian Mendel. Heuristic tool for linear cryptanalysis with applications to CAESAR candidates. In *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, pages 490–509, 2015.

[DEMS15] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Cryptanalysis of Ascon. In *Topics in Cryptology - CT-RSA 2015, The Cryptographer's Track at the RSA Conference 2015, San Francisco, CA, USA, April 20-24, 2015. Proceedings*, pages 371–387, 2015.

[DEMS16] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2. *Candidate for the CAESAR Competition. See also*, 2016. http://ascon.iaik.tugraz.at.

[DEMS21] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon v1.2: Lightweight Authenticated Encryption and Hashing. *J. Cryptol.*, 34(3):33, 2021.

[DKM$^+$17] Ashutosh Dhar Dwivedi, Milos Kloucek, Pawel Morawiecki, Ivica Nikolic, Josef Pieprzyk, and Sebastian Wójtowicz. SAT-based cryptanalysis of authenticated ciphers from the CAESAR competition. In *Proceedings of the 14th International Joint Conference on e-Business and Telecommunications (ICETE 2017) - Volume 4: SECRYPT, Madrid, Spain, July 24-26, 2017*, pages 237–246, 2017.

[DS09] Itai Dinur and Adi Shamir. Cube attacks on tweakable black box polynomials. In *Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings*, pages 278–299, 2009.

[DS11] Itai Dinur and Adi Shamir. Breaking Grain-128 with Dynamic Cube Attacks. In Antoine Joux, editor, *Fast Software Encryption - 18th International Workshop, FSE 2011, Lyngby, Denmark, February 13-16, 2011, Revised Selected Papers*, volume 6733 of *Lecture Notes in Computer Science*, pages 167–187. Springer, 2011.

[FMS01] Scott R. Fluhrer, Itsik Mantin, and Adi Shamir. Weaknesses in the Key Scheduling Algorithm of RC4. In Serge Vaudenay and Amr M. Youssef, editors, *Selected Areas in Cryptography, 8th Annual International Workshop, SAC 2001 Toronto, Ontario, Canada, August 16-17, 2001, Revised Papers*, volume 2259 of *Lecture Notes in Computer Science*, pages 1–24. Springer, 2001.

[GLR+20] Lorenzo Grassi, Gregor Leander, Christian Rechberger, Cihangir Tezcan, and Friedrich Wiemer. Weak-Key Distinguishers for AES. In Orr Dunkelman, Michael J. Jacobson Jr., and Colin O'Flynn, editors, *Selected Areas in Cryptography - SAC 2020 - 27th International Conference, Halifax, NS, Canada (Virtual Event), October 21-23, 2020, Revised Selected Papers*, volume 12804 of *Lecture Notes in Computer Science*, pages 141–170. Springer, 2020.

[GPT21] David Gerault, Thomas Peyrin, and Quan Quan Tan. Exploring Differential-Based Distinguishers and Forgeries for ASCON. Cryptology ePrint Archive, Report 2021/1103, 2021. https://ia.cr/2021/1103.

[GRW16] Faruk Göloglu, Vincent Rijmen, and Qingju Wang. On the division property of S-boxes. *IACR Cryptol. ePrint Arch.*, 2016:188, 2016.

[Haw98] Philip Hawkes. Differential-Linear Weak Key Classes of IDEA. In Kaisa Nyberg, editor, *Advances in Cryptology - EUROCRYPT '98, International Conference on the Theory and Application of Cryptographic Techniques, Espoo, Finland, May 31 - June 4, 1998, Proceeding*, volume 1403 of *Lecture Notes in Computer Science*, pages 112–126. Springer, 1998.

[HLLT20] Phil Hebborn, Baptiste Lambin, Gregor Leander, and Yosuke Todo. Lower bounds on the degree of block ciphers. In Shiho Moriai and Huaxiong Wang, editors, *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I*, volume 12491 of *Lecture Notes in Computer Science*, pages 537–566. Springer, 2020.

[HLM+20] Yonglin Hao, Gregor Leander, Willi Meier, Yosuke Todo, and Qingju Wang. Modeling for three-subset division property without unknown subset - improved cube attacks against Trivium and Grain-128aead. In Anne Canteaut and Yuval Ishai, editors, *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 I*, volume 12105 of *Lecture Notes in Computer Science*, pages 466–495. Springer, 2020.

[HSWW20] Kai Hu, Siwei Sun, Meiqin Wang, and Qingju Wang. An algebraic formulation of the division property: Revisiting degree evaluations, cube attacks, and key-independent sums. In *Advances in Cryptology - ASIACRYPT 2020 - 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7-11, 2020, Proceedings, Part I*, pages 446–476, 2020.

[HWX+17] Senyang Huang, Xiaoyun Wang, Guangwu Xu, Meiqin Wang, and Jingyuan Zhao. Conditional Cube Attack on Reduced-Round Keccak Sponge Function. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the*

Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part II, volume 10211 of *Lecture Notes in Computer Science*, pages 259–288, 2017.

[Jea16] Jérémy Jean. TikZ for Cryptographers. https://www.iacr.org/authors/tikz/, 2016.

[JLM14] Philipp Jovanovic, Atul Luykx, and Bart Mennink. Beyond $2^{c/2}$ security in Sponge-based authenticated encryption modes. In *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, pages 85–104, 2014.

[Kha19] Mustafa Khairallah. Weak Keys in the Rekeying Paradigm: Application to COMET and mixFeed. *IACR Trans. Symmetric Cryptol.*, 2019(4):272–289, 2019.

[KM07] Orhun Kara and Cevat Manap. A New Class of Weak Keys for Blowfish. In Alex Biryukov, editor, *Fast Software Encryption, 14th International Workshop, FSE 2007, Luxembourg, Luxembourg, March 26-28, 2007, Revised Selected Papers*, volume 4593 of *Lecture Notes in Computer Science*, pages 167–180. Springer, 2007.

[KMN10] Simon Knellwolf, Willi Meier, and María Naya-Plasencia. Conditional Differential Cryptanalysis of NLFSR-Based Cryptosystems. In Masayuki Abe, editor, *Advances in Cryptology - ASIACRYPT 2010 - 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings*, volume 6477 of *Lecture Notes in Computer Science*, pages 130–145. Springer, 2010.

[LAAZ11] Gregor Leander, Mohamed Ahmed Abdelraheem, Hoda AlKhzaimi, and Erik Zenner. A Cryptanalysis of PRINTcipher: The Invariant Subspace Attack. In Phillip Rogaway, editor, *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, volume 6841 of *Lecture Notes in Computer Science*, pages 206–221. Springer, 2011.

[LDW17] Zheng Li, Xiaoyang Dong, and Xiaoyun Wang. Conditional cube attack on round-reduced ASCON. *IACR Trans. Symmetric Cryptol.*, 2017(1):175–202, 2017.

[LIMS21] Fukang Liu, Takanori Isobe, Willi Meier, and Kosei Sakamoto. Weak Keys in Reduced AEGIS and Tiaoxin. *IACR Trans. Symmetric Cryptol.*, 2021(2):104–139, 2021.

[LLL21] Meicheng Liu, Xiaojuan Lu, and Dongdai Lin. Differential-Linear Cryptanalysis from an Algebraic Perspective. In Tal Malkin and Chris Peikert, editors, *Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021, Virtual Event, August 16-20, 2021, Proceedings, Part III*, volume 12827 of *Lecture Notes in Computer Science*, pages 247–277. Springer, 2021.

[LMR15] Gregor Leander, Brice Minaud, and Sondre Rønjom. A Generic Approach to Invariant Subspace Attacks: Cryptanalysis of Robin, iSCREAM and Zorro. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory* *and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 254–283. Springer, 2015.

[LTW18] Gregor Leander, Cihangir Tezcan, and Friedrich Wiemer. Searching for subspace trails and truncated differentials. *IACR Trans. Symmetric Cryptol.*, 2018(1):74–100, 2018.

[LZWW17] Yanbin Li, Guoyan Zhang, Wei Wang, and Meiqin Wang. Cryptanalysis of round-reduced ASCON. *Sci. China Inf. Sci.*, 60(3):38102, 2017.

[Men17] Bart Mennink. Weak Keys for AEZ, and the External Key Padding Attack. In Helena Handschuh, editor, *Topics in Cryptology - CT-RSA 2017 - The Cryptographers' Track at the RSA Conference 2017, San Francisco, CA, USA, February 14-17, 2017, Proceedings*, volume 10159 of *Lecture Notes in Computer Science*, pages 223–237. Springer, 2017.

[Nat19] National Institute of Standards and Technology. Lightweight Cryptography (LWC) Standardization project, 2019. https://csrc.nist.gov/projects/lightweight-cryptography.

[RHSS21] Raghvendra Rohit, Kai Hu, Sumanta Sarkar, and Siwei Sun. Misuse-Free Key-Recovery and Distinguishing Attacks on 7-Round Ascon. *IACR Trans. Symmetric Cryptol.*, 2021(1):130–155, 2021.

[TIHM17] Yosuke Todo, Takanori Isobe, Yonglin Hao, and Willi Meier. Cube attacks on non-blackbox polynomials based on division property. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III*, volume 10403 of *Lecture Notes in Computer Science*, pages 250–279. Springer, 2017.

[TLS16] Yosuke Todo, Gregor Leander, and Yu Sasaki. Nonlinear Invariant Attack - Practical Attack on Full SCREAM, iSCREAM, and Midori64. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part II*, volume 10032 of *Lecture Notes in Computer Science*, pages 3–33, 2016.

[TM16] Yosuke Todo and Masakatu Morii. Bit-based division property and application to Simon family. In Thomas Peyrin, editor, *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*, volume 9783 of *Lecture Notes in Computer Science*, pages 357–377. Springer, 2016.

[Tod15] Yosuke Todo. Structural Evaluation by Generalized Integral Property. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 287–314. Springer, 2015.

[Vie07] Michael Vielhaber. Breaking ONE.FIVIUM by AIDA an Algebraic IV Differential Attack. Cryptology ePrint Archive, Report 2007/413, 2007. https://ia.cr/2007/413.

[WHG$^+$19] Senpeng Wang, Bin Hu, Jie Guan, Kai Zhang, and Tairong Shi. MILP-aided method of searching division property using three subsets and applications. In Steven D. Galbraith and Shiho Moriai, editors, *Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part III*, volume 11923 of *Lecture Notes in Computer Science*, pages 398–427. Springer, 2019.

[WHT$^+$18] Qingju Wang, Yonglin Hao, Yosuke Todo, Chaoyun Li, Takanori Isobe, and Willi Meier. Improved division property based cube attacks exploiting algebraic properties of superpoly. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*, volume 10991 of *Lecture Notes in Computer Science*, pages 275–305. Springer, 2018.

[YLW$^+$19] Hailun Yan, Xuejia Lai, Lei Wang, Yu Yu, and Yiran Xing. New zero-sum distinguishers on full 24-round Keccak-f using the division property. *IET Inf. Secur.*, 13(5):469–478, 2019.