# Impossible Differential Cryptanalysis of Reduced-Round SKINNY

Mohamed Tolba, Ahmed Abdelkhalek, and Amr M. Youssef(✉)

Concordia Institute for Information Systems Engineering,
Concordia University, Montréal, QC, Canada
youssef@ciise.concordia.ca

**Abstract.** SKINNY is a new lightweight tweakable block cipher family proposed by Beierle *et al.* at CRYPTO 2016. SKINNY has 6 main variants where SKINNY-$n$-$t$ is a block cipher that operates on $n$-bit blocks using $t$-bit tweakey (key and tweak) where $n = 64$ or 128 and $t = n$, $2n$, or $3n$. In this paper, we present impossible differential attacks against reduced-round versions of all the 6 members of the SKINNY family in the single-tweakey model. More precisely, using an 11-round impossible differential distinguisher, we present impossible differential attacks against 18-round SKINNY-$n$-$n$, 20-round SKINNY-$n$-$2n$ and 22-round SKINNY-$n$-$3n$ ($n = 64$ or 128). To the best of our knowledge, these are the best attacks against these 6 variants in the single-tweakey model.

**Keywords:** Cryptanalysis · Impossible differential attacks · Tweakable · Block ciphers · SKINNY

## 1 Introduction

SKINNY [3] is a Substitution Permutation Network (SPN) family of tweakable lightweight block ciphers proposed at CRYPTO 2016 by Beierle *et al.* It supports two block lengths $n = 64$ or 128 and for each of them, the tweakey $t$ can be either $n$, $2n$ or $3n$. This family of ciphers inherits the recent design trend of having an SPN cipher with suboptimal internal components. More precisely, SKINNY uses a light tweakey schedule along with a round function that consists of a compact S-box and a sparse diffusion layer. However, these suboptimal components are arranged such that tight security bounds are guaranteed. Indeed, using Mixed Integer Linear Programming (MILP), the designers of SKINNY provide high security bounds against differential/linear attacks for all the SKINNY versions in both the single-tweakey and related-tweakey models. Furthermore, SKINNY has a good performance for round-based ASIC implementation as it requires a very small area using serial ASIC. Moreover, the designers of SKINNY show that its ASIC threshold implementation is very favorable to AES-128 threshold implementation [5]. Providing compact implementation and a high level of security with the existence of the tweakey was feasible by generalizing the Superposition TWEAKEY (STK) construction [7]. Lastly, being a tweakable block cipher allows SKINNY to be employed into a higher level of operating modes such as SCT [11].

M. Joye and A. Nitaj (Eds.): AFRICACRYPT 2017, LNCS 10239, pp. 117–134, 2017.
DOI: 10.1007/978-3-319-57339-7_7

The designers of SKINNY presented 16-round attacks against SKINNY-$n$-$n$ ($n = 64$ or 128) in the single-tweakey model utilizing 11-round impossible differential distinguisher. To provoke public cryptanalysis of SKINNY, they have announced a competition [2] against two particular variants of SKINNY, namely, SKINNY-64-128 and SKINNY-128-128, in which they indicated that the best known attack against SKINNY-64-128, in the single-tweakey model, is 18 rounds. As a result, a handful of third-party analysis have been published [1,10,12]. However, these attacks are in the arguably weaker attack model, the related-tweakey model, in which the attacker is assumed to have the ability to query the encryption oracle with keys that have specific relations.

In this paper, we present impossible differential attacks against reduced-round versions of all the 6 variants of SKINNY, namely, SKINNY-$n$-$n$, SKINNY-$n$-$2n$ and SKINNY-$n$-$3n$ ($n = 64$ or 128). All these attacks utilize the same 11-round impossible differential distinguisher. Then, we exploit the fact that the tweakey additions are only performed on the first two rows of the state, along with the MixColumns operation properties and the tweakey schedule relations, to extend this distinguisher by 7, 9, 11 rounds to launch key-recovery attacks in the single-tweakey model against 18, 20, 22 rounds of SKINNY-$n$-$n$, SKINNY-$n$-$2n$ and SKINNY-$n$-$3n$ ($n = 64$ or 128), respectively. Specifically, we extend the designers' 11-round impossible differential distinguisher by 3, 3 and 3 rounds above it and 4, 6 and 8 rounds below it to launch 18, 20 and 22 rounds attacks against SKINNY-$n$-$n$, SKINNY-$n$-$2n$ and SKINNY-$n$-$3n$ ($n = 64$ or 128), respectively. The time, data and memory complexities of our attacks are presented in Table 1.

**Table 1.** The time, data and memory complexities of our attacks.

| Block cipher version | # of rounds | Time | Data | Memory |
|---|---|---|---|---|
| SKINNY-64-64 | 18 | $2^{57.1}$ | $2^{47.52}$ | $2^{58.52}$ |
| SKINNY-128-128 | 18 | $2^{116.94}$ | $2^{92.42}$ | $2^{115.42}$ |
| SKINNY-64-128 | 20 | $2^{121.08}$ | $2^{47.69}$ | $2^{74.69}$ |
| SKINNY-128-256 | 20 | $2^{245.72}$ | $2^{92.1}$ | $2^{147.1}$ |
| SKINNY-64-192 | 22 | $2^{183.97}$ | $2^{47.84}$ | $2^{74.84}$ |
| SKINNY-128-384 | 22 | $2^{373.48}$ | $2^{92.22}$ | $2^{147.22}$ |

The rest of the paper is organized as follows. Section 2 provides the notations used throughout the paper and a brief description of SKINNY. In Sect. 3, we present the impossible differential distinguisher used in our attacks. The details of our attacks are presented in Sects. 4, 5 and 6, respectively. Finally, the paper is concluded in Sect. 7.

## 2 Specifications of SKINNY

The following notations are used throughout the rest of the paper:

- $TK_i$: The round tweakey used in round $i$.
- $ETK_i$: The equivalent round tweakey used in round $i$.
- $x_i$: The input to the SubCells ($SC$) operation at round $i$.
- $y_i$: The input to the AddRoundConstantTweakey ($AK$) operation at round $i$.
- $y_i^{'}$: The input to the AddRoundConstantEquivlantTweakey ($AEK$) operation at round $i$.
- $z_i$: The input to the ShiftRows ($SR$) operation at round $i$.
- $w_i$: The input to the MixColumns ($MC$) operation at round $i$.
- $x_i[j]$: The $j^{th}$ cell of $x_i$, where $0 \leq j < 16$.
- $x_i[j \cdots l]$: The cells from $j$ to $l$ of $x_i$, where $j < l$.
- $x_i[j, l]$: The cells $j$ and $l$ of $x_i$.
- $x_i[j][k]$: The $k^{th}$ bit of the $j^{th}$ cell of $x_i$.
- $x_i[j]\{k, l, m\}$: The XOR of bits $k, l, m$ of cell $j$ of $x_i$.
- $x_i[col : j]$: The four cells in column $j$, e.g., $x_i[col : 0] = x_i[0, 4, 8, 12]$.
- $x_i[SR^{-1}[col : j]]$: The four cells in column $j$ after the $SR^{-1}$ operation is applied, e.g., $x_i[SR^{-1}[col : 0]] = x_i[0, 7, 10, 13]$.
- $x_i[col : j][k, l]$: The $j^{th}$ and $l^{th}$ cells of column $j$ of $x_i$, e.g., $x_i[col : 0][0, 1] = x_i[0, 4]$.
- $\Delta x_i, \Delta x_i[j]$: The difference at state $x_i$ and cell $x_i[j]$, respectively.

The SKINNY family supports two block lengths of $n = 64$ and 128 bits. In both versions, the internal state $IS$ is represented as a $4 \times 4$ array of cells such that one cell represents a nibble (when the block length $n = 64$) and a byte (when the block length $n = 128$). While classical block ciphers have two inputs, namely the plaintext and the key, and output the ciphertext, SKINNY is a tweakable block cipher [7,9] that uses an input called the tweakey instead of the key. Then, the user has the freedom to choose which part of the tweakey to be assigned to the key and which part to be assigned to the tweak. This family of block ciphers with block length $n$ deploys three main tweakeys of lengths $t = n$ bits, $t = 2n$ bits and $t = 3n$ bits. Similar to the state, the tweakey state can be represented as $z$ $4 \times 4$ arrays of cells, i.e., we have arrays $TK1$ (in case $z = 1$), $TK1$ and $TK2$ (in case $z = 2$), $TK1$, $TK2$, and $TK3$ (in case $z = 3$).

The encryption operation proceeds as follows. First, the plaintext $m = m_0 \| m_1 \| \cdots \| m_{14} \| m_{15}$ (where $|m_i| = n/16 = s$-bit) is loaded into the internal state $IS$ row-wise as depicted in Fig. 1. Then, the tweakey input $tk = tk_0 \| tk_1 \| \cdots \| tk_{16z-1}$ (where $|tk_i|$ is $s$-bit as in the internal state) is loaded row-wise such that $TK1[i] = tk_i$ for $0 \leq i \leq 15$ (in case $z = 1$), $TK1[i] = tk_i$, $TK2[i] = tk_{16+i}$ for $0 \leq i \leq 15$ (in case $z = 2$) or $TK1[i] = tk_i$, $TK2[i] = tk_{16+i}$, $TK3[i] = tk_{32+i}$ for $0 \leq i \leq 15$ (in case $z = 3$). Finally, the internal state is updated by applying the round function $r$ times, where the number of rounds $r$ depends on the block length and the tweakey size as shown in Table 2.

As shown in Fig. 1, in each round, SKINNY applies five different operations, namely, SubCells, AddConstants, AddRoundTweakey, ShiftRows and MixColumns. The cipher does not apply whitening tweakeys. Consequently, parts of the first and last rounds do not add any security. In what follows, we describe the five different operations that are employed in each round:

**Table 2.** Number of rounds for SKINNY-$n$-$t$, with $n$-bit state and $t$-bit tweakey state.

| Block size $n$ | Tweakey size $t$ | | |
|---|---|---|---|
| | $n$ | $2n$ | $3n$ |
| 64 | 32 | 36 | 40 |
| 128 | 40 | 48 | 56 |

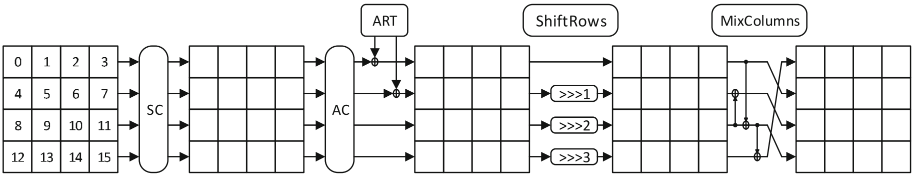

**Fig. 1.** The SKINNY round function

- SubCells ($SC$): A nonlinear bijective mapping applied on every cell of the internal state, where 4-bit (in case $n = 64$) or 8-bit (in case $n = 128$) S-boxes are applied.
- AddConstants ($AC$): A $4 \times 4$ round constant is XORed to the state. These round constants are generated using a 6-bit affine LFSR. The details of generating the round constants can be found in [3].
- AddRoundTweakey ($ART$): The first and second rows of all the tweakey arrays are XORed to the state. More precisely, for $0 \leq i \leq 7$, we have:
  - $IS[i] = IS[i] \oplus TK1[i]$, when $z = 1$,
  - $IS[i] = IS[i] \oplus TK1[i] \oplus TK2[i]$, when $z = 2$,
  - $IS[i] = IS[i] \oplus TK1[i] \oplus TK2[i] \oplus TK3[i]$, when $z = 3$.
- ShiftRows ($SR$): The rows of the state are rotated as in AES but to the right, i.e., the following permutation $P = [0, 1, 2, 3, 7, 4, 5, 6, 10, 11, 8, 9, 13, 14, 15, 12]$ is applied.
- MixColumns ($MC$): Each column in the state is multiplied by a binary matrix $M$, where $M$ and its inverse $M^{-1}$ are given as follows:

$$M = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}, \; M^{-1} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}.$$

**Tweakey Schedule.** As depicted in Fig. 2, the tweakey arrays are updated through tweakey schedule as follows. First all the tweakey arrays, i.e., *TK1* (when $z = 1$), *TK1*, *TK2* (when $z = 2$), or *TK1*, *TK2*, *TK3* (when $z = 3$) are permuted using a permutation $P_T$ such that $P_T = [9, 15, 8, 13, 10, 14, 12, 11, 0, 1, 2, 3, 4, 5, 6, 7]$. Finally, each cell in the first and second rows of *TK2*, *TK3* (when $z = 2$ or $z = 3$) is updated using the LFSR operations shown in Table 3, where $x_0$ is the LSB of the cell.

**Table 3.** The SKINNY LFSR used in the tweakey schedule, where $s$ denotes the cell size in bits.

| $TK$ | $s$ | LFSR |
|---|---|---|
| $TK2$ | 4 | $(x_3 \parallel x_2 \parallel x_1 \parallel x_0) \rightarrow (x_2 \parallel x_1 \parallel x_0 \parallel x_3 \oplus x_2)$ |
| | 8 | $(x_7 \parallel x_6 \parallel x_5 \parallel x_4 \parallel x_3 \parallel x_2 \parallel x_1 \parallel x_0) \rightarrow (x_6 \parallel x_5 \parallel x_4 \parallel x_3 \parallel x_2 \parallel x_1 \parallel x_0 \parallel x_7 \oplus x_5)$ |
| $TK3$ | 4 | $(x_3 \parallel x_2 \parallel x_1 \parallel x_0) \rightarrow (x_0 \oplus x_3 \parallel x_3 \parallel x_2 \parallel x_1)$ |
| | 8 | $(x_7 \parallel x_6 \parallel x_5 \parallel x_4 \parallel x_3 \parallel x_2 \parallel x_1 \parallel x_0) \rightarrow (x_0 \oplus x_6 \parallel x_7 \parallel x_6 \parallel x_5 \parallel x_4 \parallel x_3 \parallel x_2 \parallel x_1)$ |

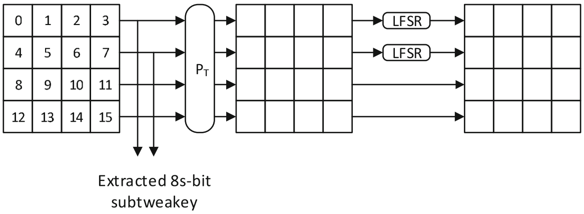

**Fig. 2.** The tweakey schedule

In our attack, we use AddKey ($AK$) operation which compromises the $AC$ and $ART$ operations. Moreover, we swap the linear operations $AK$, $MC \circ SR$, and hence we use the equivalent subtweakey $ETK$ instead of the subtweakey $TK$ such that $ETK_{r+1} = MC \circ SR(TK_r)$.

## 3 An Impossible Differential Distinguisher of SKINNY

Impossible differential cryptanalysis was proposed independently by Biham, Biryukov and Shamir [4] and Knudsen [8]. It exploits a (truncated) differential characteristic of probability exactly 0 and thus acts as a distinguisher. Then, this distinguisher is turned into a key-recovery attack by prepending and/or appending additional rounds, which are usually referred to as the analysis rounds. The keys involved in the analysis rounds which lead to the impossible differential are wrong keys and thus are excluded. Miss-in-the-Middle is the general technique used to construct impossible differentials, where a cipher $E$ is split such that $E = E_2 \circ E_1$, and we try to find two deterministic differentials, the first one covers $E_1$ and has the form $\Delta\delta \rightarrow \Delta\gamma$, and the second covers $E_2^{-1}$, and has the form $\Delta\beta \rightarrow \Delta\zeta$. When the intermediate differences $\Delta\gamma, \Delta\zeta$ do not match, the differential $\Delta\delta \rightarrow \Delta\beta$ that covers the whole cipher $E$ holds with zero probability.

The designers of SKINNY exhaustively searched for the longest truncated impossible differential that has one active cell in both $\Delta\delta$ and $\Delta\beta$. They found 16 such truncated impossible differentials where each one covers 11 rounds. They exploited one of these 16 impossible differentials, illustrated in Fig. 3, to attack 16-round SKINNY-$n$-$n$ ($n$ = 64 or 128). This distinguisher, which we reuse in our attacks, states that a pair of messages that has only one active cell at $x_3[12]$ cannot have only one active cell at $x_{14}[8]$. The reason is that the active cell

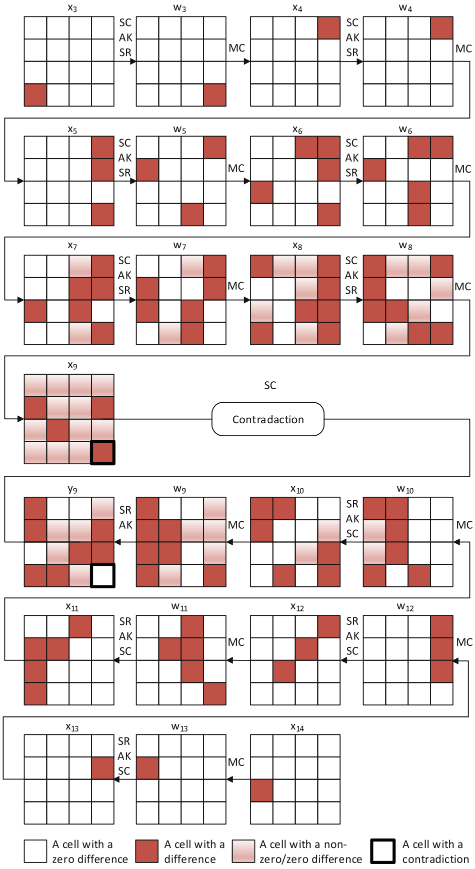

**Fig. 3.** Impossible differential distinguisher of SKINNY

$\Delta x_3[12]$ results in 4 active cells and 12 unknown cells after 6 rounds, i.e., at state $x_9$. From the other side, the active cell $\Delta x_{14}[8]$ results in 4 inactive cells, 5 unknown cells and 7 active cells at state $Y_9$ contradicting with the forward differential at $\Delta y_9[15]$.

Our attacks depend on the following proposition:

**Proposition 1** *(Differential Property of the S-box)***.** *Given two nonzero differences $\Delta i$ and $\Delta o$ in $\mathbb{F}16$ or $\mathbb{F}256$, the equation: $S(x) + S(x + \Delta i) = \Delta o$ has one solution on average. This property also applies to $S^{-1}$.*

All our attacks use the same 11-round distinguisher, have 3 analysis rounds on its top. They, however, differ in the analysis rounds appended below it. In what follows, we describe our attack against SKINNY-64-128 in details and then mention only the main differences for the other attacks.

# 4 Impossible Differential Key-Recovery Attack on 20-Round SKINNY-$n$-$2n$ ($n = 64$ or $128$)

## 4.1 Impossible Differential Key-Recovery Attack on SKINNY-64-128

In this section, we present the first published attack on 20-round SKINNY-64-128 in the single-tweakey model. We use the notion of data structures to generate enough pairs of messages to launch the attack. In the first three rounds, we use the equivalent tweakey $ETK$ instead of the tweakey $TK$. Therefore, the first round has no tweakey, and hence we can build our structures at $y_1^{'}$. Then, we propagate it backward linearly through $MC^{-1}, SR^{-1}$, and $SC^{-1}$ to obtain the corresponding plaintexts. Our utilized structure takes all the possible values in 7 nibbles $y_1^{'}[3, 4, 5, 6, 9, 11, 14]$ while the remaining nibbles take a fixed value. Thus, one structure generates $2^{4\times 7} \times (2^{4\times 7} - 1)/2 \approx 2^{55}$ possible pairs. Hence, we have $2^{55}$ possible pairs of messages satisfying the plaintext differences. In addition, we utilize the following pre-computation tables in order to efficiently extract/filter the (equivalent) tweakey nibbles corresponding to the active state nibbles involved in the analysis rounds, where the table $H_l\{(E)TK_i[\mathbb{S}]\}$ (also referred to as $H_l$) is used to extract/filter the (equivalent) tweakey used in round $i$ at cells belonging to the set $\mathbb{S}$ and $H^*$ is computed once and used to extract all the tweakey nibbles of the last analysis round and those corresponding to column 1 in round 18.

$\boldsymbol{H_1\{TK_{18}[2,6]\}}$**:** For all the $2^{24}$ possible values of $\Delta z_{17}[SR^{-1}[col : 2][0,1]]$, $z_{17}[SR^{-1}[col : 2]]$, compute $\Delta y_{18}[col : 2], y_{18}[col : 2]$. Then, store $\Delta z_{17}[SR^{-1}[col : 2][0,1]]$, $z_{17}[SR^{-1}[col : 2]], y_{18}[col : 2][0,1]$ in $H_1$ indexed by $\Delta y_{18}[col : 2], y_{18}[col : 2][2,3]$. $H_1$ has $2^{24}$ rows and on average about $2^{24}/2^{24} = 1$ value in each row.

$\boldsymbol{H_2\{TK_{18}[0,4]\}}$**:** For all the $2^{28}$ possible values of $\Delta z_{17}[SR^{-1}[col : 0][0,2,3]]$, $z_{17}[SR^{-1}[col : 0]]$, compute $\Delta y_{18}[col : 0], y_{18}[col : 0]$. Then, store $\Delta z_{17}[SR^{-1}[col : 0][0,2,3]]$, $z_{17}[SR^{-1}[col : 0]], y_{18}[col : 0][0,1]$ in $H_2$ indexed by $\Delta y_{18}[col : 0]$, $y_{18}[col : 0][2,3]$. $H_2$ has $2^{24}$ rows and on average about $2^{28}/2^{24} = 2^4$ values in each row.

$\boldsymbol{H_3\{TK_{18}[3,7]\}}$**:** For all the $2^{28}$ possible values of $\Delta z_{17}[SR^{-1}[col : 3][0,1,3]]$, $z_{17}[SR^{-1}[col : 3]]$, compute $\Delta y_{18}[col : 3], y_{18}[col : 3]$. Then, store $\Delta z_{17}[SR^{-1}[col : 3][0,1,3]]$, $z_{17}[SR^{-1}[col : 3]], y_{18}[col : 3][0,1]$ in $H_3$ indexed by $\Delta y_{18}[col : 3]$, $y_{18}[col : 3][2,3]$. $H_3$ has $2^{24}$ rows and on average about $2^{28}/2^{24} = 2^4$ values in each row.

$\boldsymbol{H_4\{TK_{17}[0,4]\}}$**:** For all the $2^{20}$ possible values of $\Delta z_{16}[SR^{-1}[col : 0][0]]$, $z_{16}[SR^{-1}[col : 0]]$, compute $\Delta y_{17}[col : 0][0,1,3]$, $y_{17}[col : 0]$. Then, store $\Delta z_{16}$ $[SR^{-1}[col : 0][0]]$, $z_{16}[SR^{-1}[col : 0]], y_{17}[col : 0][0,1]$ in $H_4$ indexed by $\Delta y_{17}[col : 0][0,1,3], y_{17}[col : 0][2,3]$. $H_4$ has $2^{20}$ rows and on average about $2^{20}/2^{20} = 1$ value in each row.

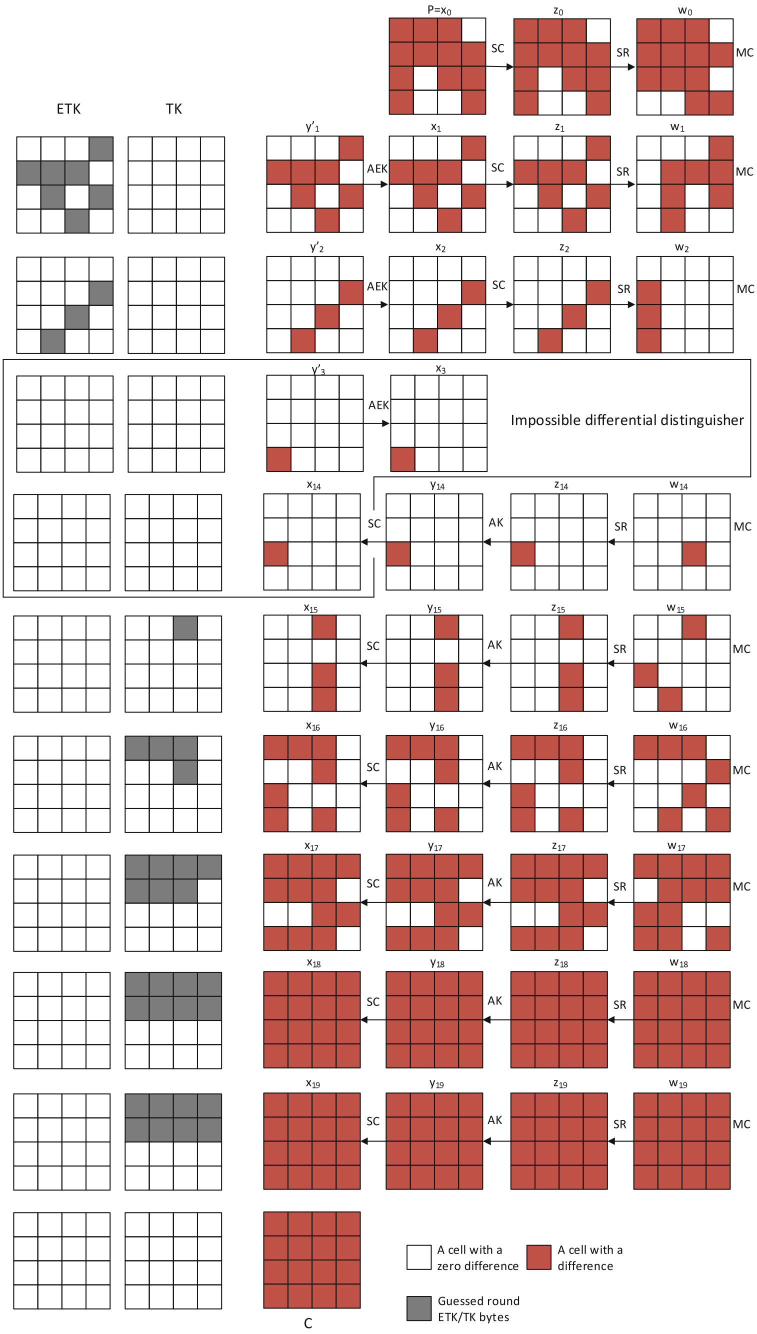

**Fig. 4.** Impossible differential attack on 20-round SKINNY-$n$-$2n$

$\boldsymbol{H_5\{TK_{17}[2,3,6]\}}$**:** From the properties of the MixColumns, we have $\Delta x_{16}[0] = \Delta x_{16}[8] = \Delta x_{16}[12] = \Delta w_{15}[8]$. Therefore, for all the $2^{40}$ possible values for $\Delta x_{16}[8]$, $x_{16}[8,12], \Delta w_{16}[2,7], w_{16}[2,6,14], x_{17}[3,11]$, compute $w_{16}[10,15], \Delta y_{17}[2,3,6,10,11,14], y_{17}[2,3,6,10,11,14,15]$ such that $y_{17}[15] = SC([w_{16}[15] \oplus x_{17}[3])$, from the MixColumns operation. Then, store $\Delta z_{16}[SR^{-1}[col:2][0,2]], \Delta z_{16}[ SR^{-1}[col:3][1,3]]$, $z_{16}[SR^{-1}[col:2]]$, $z_{16}[SR^{-1}[col:3][3]]$, $y_{17}[2,3,6]$ in $H_5$ indexed by $\Delta y_{17}[2,3,6,10, 11,14], y_{17}[10,11,14,15]$. $H_5$ has $2^{40}$ rows and on average about $2^{40}/2^{40} = 1$ value in each row.

$\boldsymbol{H_6\{TK_{17}[1,5]\}}$**:** For all the $2^{24}$ possible values of $\Delta z_{16}[SR^{-1}[col:1][0,3]], z_{16}[SR^{-1}[col:1]]$, compute $\Delta y_{17}[col:1][0,1,3], y_{17}[col:1]$. Then, store $\Delta z_{16}[SR^{-1}[col:1][0,3]]$, $z_{16}[SR^{-1}[col:1]], y_{17}[col:1][0,1]$ in $H_6$ indexed by $\Delta y_{17}[col:1][0,1,3], y_{17}[col:1][2,3]$. $H_6$ has $2^{20}$ rows and on average about $2^{24}/2^{20} = 2^4$ values in each row.

$\boldsymbol{H_7\{TK_{16}[0]\}}$**:** For all the $2^{20}$ possible values of $\Delta z_{15}[SR^{-1}[col:0][2]], z_{15}[SR^{-1}[col:0]]$, compute $\Delta y_{16}[col:0][0,2,3]$, $y_{16}[col:0]$. Then, store $\Delta z_{15}[SR^{-1}[col:0][2]]$, $z_{15}[SR^{-1}[col:0]], y_{16}[col:0][0]$ in $H_7$ indexed by $\Delta y_{16}[col:0][0,2,3], y_{16}[col:0][2,3]$. $H_7$ has $2^{20}$ rows and on average about $2^{20}/2^{20} = 1$ value in each row.

$\boldsymbol{H_8\{TK_{16}[2]\}}$**:** For all the $2^{20}$ possible values of $\Delta z_{15}[SR^{-1}[col:2][0]], z_{15}[SR^{-1}[col:2]]$, compute $\Delta y_{16}[col:2][0,1,3]$, $y_{16}[col:2]$. Then, store $\Delta z_{15}[SR^{-1}[col:2][0]], z_{15}[SR^{-1}[col:2]], y_{16}[col:2][0,1]$ in $H_8$ indexed by $\Delta y_{16}[col:2][0,1,3], y_{16}[col:2][2,3]$. $H_8$ has $2^{20}$ rows and on average about $2^{20}/2^{20} = 1$ value in each row.

$\boldsymbol{H_9\{TK_{15}[2]\}}$**:** From the properties of the MixColumns, we have $\Delta x_{15}[2] = \Delta x_{15}[10] = \Delta x_{15}[14] = \Delta w_{14}[10]$. Therefore, for all the $2^4$ possible differences for $\Delta x_{15}[2,10]$, $2^8$ possible values of $x_{15}[2,10]$ and $2^4$ possible values of $TK_{15}[2]$, compute $\Delta z_{15}[2,10], z_{15}[2,10]$. Then, store $\Delta z_{15}[2]$ in $H_9$ indexed by $\Delta z_{15}[2,10], z_{15}[2,10]$, $TK_{15}[2]$. $H_9$ has $2^{20}$ rows and on average about $2^{16}/2^{20} = 2^{-4}$ values in each row.

$\boldsymbol{H_{10}\{ETK_1[4,11,14]\}}$**:** For all the $2^{12}$ possible differences of $\Delta w_1[5,9,13]$, we have only $2^4$ valid differences that have exactly one difference in $\Delta y'_2[13]$ and 3 zero differences in $\Delta y'_2[1,5,9]$. Therefore, for all the $2^4$ possible differences of $\Delta w_1[5,9,13]$, $2^{12}$ possible values of $w_1[5,9,13]$ and $2^8$ possible values of $ETK_1[4,14]$, compute $\Delta y'_1[4,14], y'_1[4,14], \Delta x_1[11], x_1[11]$. Then, store $\Delta w_1[5,9, 13], w_1[5,9,13], x_1[11]$ in $H_{10}$ indexed by $\Delta y'_1[4,14], y'_1[4,14], \Delta x_1[11]$, $ETK_1[4,14]$. $H_{10}$ has $2^{28}$ rows and on average about $2^{24}/2^{28} = 2^{-4}$ values in each row.

$\boldsymbol{H_{11}\{ETK_1[3,6,9]\}}$**:** For all the $2^{12}$ possible differences of $\Delta w_1[3,7,11]$, we have only $2^4$ valid differences that have exactly one difference in $\Delta y'_2[7]$ and 3 zero differences in $\Delta y'_2[3,11,15]$. Therefore, for all the $2^4$ possible differences of $\Delta w_1[3,7,11]$, $2^{12}$ possible values of $w_1[3,7,11]$ and $2^4$ possible values of $ETK_1[6]$, compute $\Delta y'_1[6], y'_1[6], \Delta x_1[3,9], x_1[3,9]$. Then, store $\Delta w_1[3,7,11], w_1[3,7,11], x_1[ 3,9]$ in

$H_{11}$ indexed by $\Delta x_1[3,9], \Delta y_1^{'}[6], y_1^{'}[6]$,$ETK_1[6]$. $H_{11}$ has $2^{20}$ rows and on average about $2^{20}/2^{20} = 1$ value in each row.

$\boldsymbol{H_{12}\{TK_{16}[1]\}}$**:** For all the $2^8$ possible values of $\Delta x_{16}[1], x_{16}[1]$, compute $\Delta y_{16}[1], y_{16}[1]$. Then, store $y_{16}[1]$ in $H_{12}$ indexed by $\Delta y_{16}[1]$. $H_{12}$ has $2^4$ rows and on average about $2^8/2^4 = 2^4$ values in each row.

$\boldsymbol{H_{13}\{ETK_1[1,5]\}}$**:** For all the $2^{16}$ possible values of $\Delta w_1[6], w_1[1,6], ETK_1[1,5]$ ($ETK_1[1] = ETK_1[5]$, see Appendix A in the full version of this paper [13]), compute $\Delta y_1^{'}[5], y_1^{'}[1,5]$. Then, store $\Delta w_1[6], w_1[1,6]$ in $H_{13}$ indexed by $\Delta y_1^{'}[5], y_1^{'}[1,5]$, $ETK_1$ [1]. $H_{13}$ has $2^{16}$ rows and on average about $2^{16}/2^{16} = 1$ value in each row.

$\boldsymbol{H_{14}\{ETK_2[7,10,13]\}}$**:** From the properties of the MixColumns, we have $\Delta w_2[4] = \Delta w_2[8] = \Delta w_2[12] = \Delta y_3^{'}[12]$. Therefore, for all the $2^4$ possible differences for $\Delta w_2[4,8,12]$, $2^{12}$ possible values of $w_2[4,8,12]$ and $2^{12}$ possible values of $ETK_2[7,10,13]$, compute $\Delta y_2^{'}[7,10,13], y_2^{'}[7,10,13]$. Then, store $\Delta y_2^{'}[10]$ in $H_{14}$ indexed by $\Delta y_2^{'}[7,10,13], y_2^{'}[7,13]$,$ETK_2[7,10,13]$. $H_{14}$ has $2^{32}$ rows and on average about $2^{28}/2^{32} = 2^{-4}$ value in each row.

$\boldsymbol{H^*}$**:** For all the $2^{32}$ possible values of $\Delta z_i[SR^{-1}[col:j]], z_i[SR^{-1}[col:j]]$, compute $\Delta y_{i+1}[col:j], y_{i+1}[col:j]$. Then, store $\Delta z_i[SR^{-1}[col:j]], z_i[SR^{-1}[col:j]], y_{i+1}[col:j][0,1]$ in $H^*$ indexed by $\Delta y_{i+1}[col:j], y_{i+1}[col:j][2,3]$. $H^*$ has $2^{24}$ rows and on average about $2^{32}/2^{24} = 2^8$ values in each row.

Instead of guessing the tweakey nibbles involved in the analysis rounds as in the general approach of impossible differential attacks, we use the above mentioned pre-computation tables to deduce the tweakey nibbles that lead a specific pair of plaintext/ciphertext to the impossible differential and thus should be excluded. The details of our attack are as follows:

1. Generate $2^m$ structures as described above. Therefore, we have $2^{m+55}$ pairs of messages generated using $2^{m+28}$ messages. Then, ask the encryption oracle for their corresponding ciphertexts and decrypt them partially over $MC^{-1}, SR^{-1}$ to compute $z_{19}$.
2. Determine the number of possible values of $TK_{19}[0:7]$ that satisfy the last round by performing the following steps for all the message pairs:
   (a) Access $H^*$ for $i = 18, j = 0$ and compute $TK_{19}[0,4]$ such that $TK_{19}[0,4] = y_{19}[0,4] \oplus z_{19}[0,4]$[1]. Therefore, we have $2^8$ possible tweakeys for $TK_{19}[0,4]$.
   (b) Access $H^*$ for $i = 18, j = 1$ and compute $TK_{19}[1,5]$ such that $TK_{19}[1,5] = y_{19}[1,5] \oplus z_{19}[1,5]$. Therefore, we have $2^{8+8=16}$ possible tweakeys for $TK_{19}[0,1,4,5]$.

[1] $TK_{19}[0,4] = y_{19}[0,4] \oplus z_{19}[0,4]$ means that $TK_{19}[0] = y_{19}[0] \oplus z_{19}[0], TK_{19}[4] = y_{19}[4] \oplus z_{19}[4]$.

  (c) Access $H^*$ for $i = 18, j = 2$ and compute $TK_{19}[2,6]$ such that $TK_{19}[2,6] = y_{19}[2,6] \oplus z_{19}[2,6]$. Therefore, we have $2^{16+8=24}$ possible tweakeys for $TK_{19}[0,1,2,4,5,6]$.
  (d) Access $H^*$ for $i = 18, j = 3$ and compute $TK_{19}[3,7]$ such that $TK_{19}[3,7] = y_{19}[3,7] \oplus z_{19}[3,7]$. Therefore, we have $2^{24+8=32}$ possible tweakeys for $TK_{19}[0:7]$.

3. Determine the number of possible values of $TK_{18}[0:7]$ that satisfy the next to last round by performing the following steps for all the message pairs and remaining tweakeys that satisfy the path until now:

   (a) Access $H_1$ and compute $TK_{18}[2,6]$ such that $TK_{18}[2,6] = y_{18}[2,6] \oplus z_{18}[2,6]$. Therefore, we have $2^{32}$ possible tweakeys for $TK_{19}[0:7], TK_{18}$ [2 ,6].
   (b) Access $H_2$ and compute $TK_{18}[0,4]$ such that $TK_{18}[0,4] = y_{18}[0,4] \oplus z_{18}[0,4]$. Therefore, we have $2^{32+4=36}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0,2,4,6]$.
   (c) Access $H_3$ and compute $TK_{18}[3,7]$ such that $TK_{18}[3,7] = y_{18}[3,7] \oplus z_{18}[3,7]$. Therefore, we have $2^{36+4=40}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0,2,3,4,6,7]$.
   (d) Access $H^*$ for $i = 17, j = 1$ and compute $TK_{18}[1,5]$ such that $TK_{18}[1,5] = y_{18}[1,5] \oplus z_{18}[1,5]$. Therefore, we have $2^{40+8=48}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7]$.

4. Determine the number of possible values of $TK_{17}[0:6]$ that satisfy the eighteenth round by performing the following steps for all the message pairs and remaining tweakeys that satisfy the path until now:

   (a) Access $H_4$ and compute $TK_{17}[0,4]$ such that $TK_{17}[0,4] = y_{17}[0,4] \oplus z_{17}[0,4]$. Therefore, we have $2^{48}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0,4]$.
   (b) Access $H_5$ and compute $TK_{17}[2,3,6]$ such that $TK_{17}[2,3,6] = y_{17}[2,3,6] \oplus z_{17}[2,3,6]$. Therefore, we have $2^{48}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0,2,3,4,6]$.
   (c) Access $H_6$ and compute $TK_{17}[1,5]$ such that $TK_{17}[1,5] = y_{17}[1,5] \oplus z_{17}[1,5]$. Therefore, we have $2^{48+4=52}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0:6]$.

5. Determine the number of possible values of $TK_{16}[0,2]$ that satisfy the seventeenth round by performing the following steps for all the message pairs and remaining tweakeys that satisfy the path until now:

   (a) Access $H_7$ and compute $TK_{16}[0]$ such that $TK_{16}[0] = y_{16}[0] \oplus z_{16}[0]$. Therefore, we have $2^{52}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0:6], TK_{16}[0]$.
   (b) Access $H_8$ and compute $TK_{16}[2]$ such that $TK_{16}[2] = y_{16}[2] \oplus z_{16}[2]$. Therefore, we have $2^{52}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0:6], TK_{16}[0,2]$[2].

[2] Note that instead of having $TK_{16}[6]$ that lead to the impossible differential distinguisher, we have $x_{16}[6]$ that result in the same impossible differential distinguisher.

6. The knowledge of $TK_{19}[6]$ and $TK_{17}[4]$ enables us to deduce $TK_{15}[2]$ (see Appendix A in [13]). Hence, we determine the number of possible tweakey values that satisfy the sixteenth round by performing the following steps for all the message pairs and remaining tweakeys that satisfy the path until now:

   (a) Access $H_9$; and we will find $2^{-4}$ possible values in each row, i.e., we have 4-bit filter on the remaining tweakeys. Therefore, we have $2^{52-4=48}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0:6], TK_{16}[0,2]\ TK_{15}[2]$.

7. The knowledge of $TK_{18}[2,4]$ and $TK_{16}[0,2]$ enables us to deduce $ETK_1[4,6,14]$[3] (see Appendix A in [13]). Hence, we determine the number of possible values for $ETK_1[3,9,11]$ that satisfy the second round by performing the following steps for all the message pairs and remaining tweakeys that satisfy the path until now:

   (a) Access $H_{10}$ and compute $ETK_1[11]$ such that $ETK_1[11] = y_1^{'}[11] \oplus x_1[11]$; we will find $2^{-4}$ possible values in each row, i.e., we have 4-bit filter on the remaining tweakeys. Therefore, we have $2^{48-4=44}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0:6], TK_{16}[0,2], TK_{15}[2], ETK_1[4,6,11,14]$.
   (b) Access $H_{11}$ and compute $ETK_1[3,9]$ such that $ETK_1[3,9] = y_1^{'}[3,9] \oplus x_1[3,9]$. Therefore, we have $2^{44}$ possible tweakeys for $TK_{19}[0:7]$,$TK_{18}[0:7], TK_{17}[0:6], TK_{16}[0,2], TK_{15}[2]$, $ETK_1[3,4,6,9,11,14]$.

8. Determine the number of possible values for $TK_{16}[1]$ that satisfy the seventeenth round by performing the following steps for all the message pairs and remaining tweakeys that satisfy the path until now:

   (a) Access $H_{12}$ and compute $TK_{16}[1]$ such that $TK_{16} = y_{16}[1] \oplus z_{16}[1]$. Therefore, we have $2^{44+4=48}$ possible tweakeys for $TK_{19}[0:7]$,$TK_{18}[0:7], TK_{17}[0:6]\ , TK_{16}[0,1,2], TK_{15}[2]$, $ETK_1[3,4,6,9,11,14]$.

9. The knowledge of $TK_{18}[0]$ and $TK_{16}[1]$ enables us to deduce $ETK_1[1,5]$ (see footnote 3) (see Appendix A in [13]). Hence, we determine the number of possible tweakey values that satisfy the second round by performing the following steps for all the message pairs and remaining tweakeys that satisfy the path until now:

   (a) Access $H_{13}$ and we will find 1 possible value in each row. Therefore, we have $2^{48}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0:6]$ , $TK_{16}[0,1,2], TK_{15}[2]$, $ETK_1[1,3,4,5,6,9,11,14]$,.

10. The knowledge of $TK_{19}[0,3,7]$ and $TK_{17}[1,3,5]$ enables us to deduce $ETK_2[7,\ 10,13]$ (see Appendix A in [13]). Hence, we determine the number of possible tweakey values that satisfy the third round by performing the following steps for all the message pairs and remaining tweakeys that satisfy the path until now:

[3] Note that $ETK_1[6] = ETK_1[14]$ and $ETK_1[1] = ETK_1[5]$.

(a) Access $H_{14}$ and we will find $2^{-4}$ possible values in each row. Therefore, we have $2^{48-4=44}$ possible tweakeys for $TK_{19}[0:7]$, $TK_{18}[0:7], TK_{17}[0:6]$, $TK_{16}[0,1,2], TK_{15}[2]$, $ETK_1[1,3,4,5,6,9,11,14]$, $ETK_2[7,10,13]$.

**Attack Complexity.** As depicted in Fig. 4, we have 38 tweakey nibbles that are involved in the analysis rounds. Thanks to the tweakey schedule, these 38 nibbles take only $2^{116}$ possible values (see Appendix A in [13]). For each of the $2^{m+55}$ message pairs, we remove, on average, $2^{44}$ out of $2^{116}$ possible values of these tweakey nibbles. Therefore, the probability that a wrong tweakey is not discarded with one pair is $1-2^{44-116}=1-2^{-72}$. Hence, after processing all the $2^{m+55}$ pairs, we have $2^{116}(1-2^{-72})^{2^{m+55}} \approx 2^{116} \times (e^{-1})^{2^{m+55-72}} \approx 2^{116} \times 2^{-1.4 \times 2^{m-17}}$ remaining candidates for 116-bit of the tweakey. In order to determine the optimal value of $m$ that leads to the best computational complexity, we evaluate the computational complexity of the attack as a function of $m$, as illustrated in Table 4. Similar to AES [6], the SKINNY round function can be implemented using 16 table lookups. As seen from Table 4, steps 5(a), 5(b) and 6(a) dominate the time complexity of the attack, and hence in order to optimize the time complexity of the attack we choose $m = 19.69$. Consequently, we have $2^{107}$ remaining tweakey candidates for the 116-bit of the tweakey. Therefore, the tweakey can be recovered by exhaustively searching the $2^{107}$ remaining tweakey candidates with $2^{12}$ remaining tweakey bits, that are not involved in the attack, using 2 plaintext/ciphertext pairs. Therefore, the total time complexity of the attack is $2 \times 2^{107} \times 2^{12} + 2^{120.15} = 2^{121.08}$ encryptions. The data complexity of the attack can be determined from step 1 in which we generate $2^{m=19.69}$ structures. Hence, the data complexity of the attack is $2^{19.69+28=47.69}$ chosen plaintexts. The memory complexity of the attack is dominated by the memory that is required to store $2^{m+55=74.69}$ pairs to exclude the wrong tweakeys, hence, it is $2^{74.69}$.

### 4.2 Impossible Differential Key-Recovery Attack on SKINNY-128-256

The only difference between SKINNY-64-128 and SKINNY-128-256 is the tweakey schedule, more precisely, the LFSR operation. The above attack on SKINNY-64-128 can be applied on SKINNY-128-256 while only considering that the cell size $s = 8$. Therefore, one structure can generate $2^{111}$ pairs with $2^{56}$ chosen plaintexts. According to the tweakey schedule, the 38 bytes involved in the attack have $2^{232}$ possible values (see Appendix B in the full version of this paper [13]). In this attack, we exclude, on overage, $2^{88}$ out of $2^{232}$ possible values of the involved tweakey bytes for every message pair. Hence, the probability that one wrong tweakey is not discarded is $1-2^{88-232}=1-2^{-144}$. Therefore, we have $2^{232} \times (1-2^{-144})^{2^{m+111}} \approx 2^{232} \times (e^{-1})^{2^{m+111-144}} \approx 2^{232} \times 2^{-1.4 \times 2^{m-33}}$ remaining candidates for 232-bit of the tweakey bytes, after processing all the message pairs. In order to optimize the time complexity of the attack, we choose $m = 36.1$. Consequently, we have $2^{220}$ remaining candidates for 232-bit of the tweakey, and hence the tweakey can be recovered by exhaustively searching the remaining candidates with $2^{24}$ possible values, for the 24 bits of the tweakey that

**Table 4.** Time complexity of the different steps of the attack on 20-round SKINNY-64-128, where NT denotes the number of tweakeys to be excluded.

| Step | Time complexity (in 20-round encryptions) | NT | $m = 19.69$ |
|---|---|---|---|
| 1 | $2^{m+28}$ | - | $2^{47.69}$ |
| 2(a) | $2^{m+55} \times \frac{1}{16 \times 20} \approx 2^{m+46.68}$ | $2^{8}$ | $2^{66.37}$ |
| 2(b) | $2^{m+55} \times 2^{8} \times \frac{1}{16 \times 20} \approx 2^{m+54.68}$ | $2^{16}$ | $2^{74.37}$ |
| 2(c) | $2^{m+55} \times 2^{16} \times \frac{1}{16 \times 20} \approx 2^{m+62.68}$ | $2^{24}$ | $2^{82.37}$ |
| 2(d) | $2^{m+55} \times 2^{24} \times \frac{1}{16 \times 20} \approx 2^{m+70.68}$ | $2^{32}$ | $2^{90.37}$ |
| 3(a) | $2^{m+55} \times 2^{32} \times \frac{1}{16 \times 20} \approx 2^{m+78.68}$ | $2^{32}$ | $2^{98.37}$ |
| 3(b) | $2^{m+55} \times 2^{32} \times \frac{1}{16 \times 20} \approx 2^{m+78.68}$ | $2^{36}$ | $2^{98.37}$ |
| 3(c) | $2^{m+55} \times 2^{36} \times \frac{1}{16 \times 20} \approx 2^{m+82.68}$ | $2^{40}$ | $2^{102.37}$ |
| 3(d) | $2^{m+55} \times 2^{40} \times \frac{1}{16 \times 20} \approx 2^{m+86.68}$ | $2^{48}$ | $2^{106.37}$ |
| 4(a) | $2^{m+55} \times 2^{48} \times \frac{1}{16 \times 20} \approx 2^{m+94.68}$ | $2^{48}$ | $2^{114.37}$ |
| 4(b) | $2^{m+55} \times 2^{48} \times \frac{2}{16 \times 20} \approx 2^{m+95.68}$ | $2^{48}$ | $2^{115.37}$ |
| 4(c) | $2^{m+55} \times 2^{48} \times \frac{1}{16 \times 20} \approx 2^{m+94.68}$ | $2^{52}$ | $2^{114.37}$ |
| 5(a) | $2^{m+55} \times 2^{52} \times \frac{1}{16 \times 20} \approx 2^{m+98.68}$ | $2^{52}$ | $2^{118.37}$ |
| 5(b) | $2^{m+55} \times 2^{52} \times \frac{1}{16 \times 20} \approx 2^{m+98.68}$ | $2^{52}$ | $2^{118.37}$ |
| 6(a) | $2^{m+55} \times 2^{52} \times \frac{1}{16 \times 20} \approx 2^{m+98.68}$ | $2^{48}$ | $2^{118.37}$ |
| 7(a) | $2^{m+55} \times 2^{48} \times \frac{1}{16 \times 20} \approx 2^{m+94.68}$ | $2^{44}$ | $2^{114.37}$ |
| 7(b) | $2^{m+55} \times 2^{44} \times \frac{1}{16 \times 20} \approx 2^{m+90.68}$ | $2^{44}$ | $2^{110.37}$ |
| 8(a) | $2^{m+55} \times 2^{44} \times \frac{1}{16 \times 20} \approx 2^{m+90.68}$ | $2^{48}$ | $2^{110.37}$ |
| 9(a) | $2^{m+55} \times 2^{48} \times \frac{1}{16 \times 20} \approx 2^{m+94.68}$ | $2^{48}$ | $2^{114.37}$ |
| 10(a) | $2^{m+55} \times 2^{48} \times \frac{1}{16 \times 20} \approx 2^{m+94.68}$ | $2^{44}$ | $2^{114.37}$ |

are not involved in the attack, using 2 plaintext/ciphertext pairs. Therefore, the total time complexity of the attack is $2 \times 2^{220} \times 2^{24} + 2^{36.1+111} \times 2^{104} \times \frac{3}{16\times 20}$[4]$= 2^{245} + 2^{244.36} = 2^{245.72}$. The data complexity of the attack is $2^{m+56=92.1}$ chosen plaintexts; and the memory complexity is dominated by storing $2^{m+111=147.1}$ message pairs.

[4] The second term is computed from step 5(a), 5(b) and 6(a).

## 5 Impossible Differential Key-Recovery Attack on 18-Round SKINNY-*n*-*n* ($n = 64$ or 128)

The only difference between SKINNY-64-64 and SKINNY-128-128 is the cell size $s$, where $s = 4$ (resp. $s = 8$) in case of SKINNY-64-64 (resp. SKINNY-128-128). Therefore, we present the steps of the two attacks concurrently as a function of $s$. This attack is applicable to the first 18 rounds of the 20-round attack on SKINNY-$n$-$2n$, i.e., the ciphertext $c = x_{18}$. Therefore, we use the same steps used in the previous attack from step 4 to the end and the same precomputation tables from $H_4$ to the end with the following modifications:

- Each structure can generate $2^{7\times s} \times 2^{7\times s-1} = 2^{14\times s-1}$ with $2^{7\times s}$ chosen plaintexts. Then, to apply the attack we take $2^m$ structures to generate $2^{m+14\times s-1}$ pairs, but we have 4 s-bit filter in the transition over $MC^{-1}$ from the ciphertext to $w_{17}$. Therefore, we have $2^{m+14\times s-1-4\times s=m+10\times s-1}$ remaining pairs to launch the attack.
- The number of rows and entries in each table will be represented as a function of $s$. For example, $H_6$ has $2^{5\times s}$ rows; and in each row, we have $2^s$ entries.
- The modifications of the number of tweakeys to be excluded from step 4 to the end are presented in Table 5.
- The relation of the tweakey cells can be found in Appendix C in the full version of this paper [13].

**Attack Complexity.** We have 22 tweakey cells that are involved in the analysis rounds where these 22 tweakey cells have only $2^{13\times s}$ possible values (see Appendix C in [13]). The probability that one wrong tweakey is not discarded with one pair is $1 - 2^{-s-13\times s} = 1 - 2^{-14\times s}$. Hence, after processing all the $2^{m+10\times s-1}$ pairs, we have $2^{13\times s}(1 - 2^{-14\times s})^{2^{m+10\times s-1}} \approx 2^{13\times s} \times (e^{-1})^{2^{m+10\times s-1-14\times s}} \approx 2^{13\times s} \times 2^{-1.4\times 2^{m-4\times s-1}}$ remaining candidates for $13 \times s$-bit of the tweakey. Steps 5(a), 5(b) and 6(a) dominate the time complexity of the attack, as seen from Table 5, and hence in order to optimize the time complexity of the attack we choose $m = 19.52$ (resp. $m = 36.42$) in case of SKINNY-64-64 (resp. SKINNY-128-128). Consequently, we have $2^{44}$ (resp. $2^{89}$) remaining tweakey candidates for the 52-bit (resp. 104-bit) of the tweakey. Therefore, the tweakey can be recovered by exhaustively searching the $2^{44}$ (resp. $2^{89}$) remaining tweakey candidates with $2^{12}$ (resp. $2^{24}$) for the other tweakey bits, that are not involved in the attack, using 1 plaintext/ciphertext pair. Therefore, the total time complexity of the attack is $2^{44} \times 2^{12} + 2^{56.14} = 2^{57.1}$ (resp. $2^{89} \times 2^{24} + 2^{116.84} = 2^{116.94}$) encryptions in case of SKINNY-64-64 (resp. SKINNY-128-128). The data complexity of the attack can be determined from step 1 in which we generate $2^{m=19.52}$ (resp. $2^{m=36.42}$) structures. Hence, the data complexity of the attack is $2^{19.52+28=47.52}$ (resp. $2^{36.42+56=92.42}$) chosen plaintexts in case of SKINNY-64-64 (resp. SKINNY-128-128). The memory complexity is dominated by the memory required to store the $2^{58.52}$ (resp. $2^{115.42}$) pairs after the ciphertext filtration and is estimated to be $2^{58.52}$ (resp. $2^{115.42}$) in case of SKINNY-64-64 (resp. SKINNY-128-128).

## 6 Impossible Differential Key-Recovery Attack on 22-Round SKINNY-$n$-$3n$ ($n = 64$ or $128$)

SKINNY-64-192 differs from SKINNY-128-384 in the cell size $s$ and the tweakey schedule. As the tweakey schedule does not influence the attack procedure, we present the two attacks as a function of $s$. The 20-round attack on SKINNY-$n$-$2n$ ($n = 64$ or 128) can be extended to 22-round attack on SKINNY-$n$-$3n$ ($n = 64$ or 128) by appending 2 rounds, i.e., the ciphertext $c = x_{22}$. Therefore, we can use the same attack procedures of SKINNY-$n$-$2n$ ($n = 64$ or 128) to attack SKINNY-$n$-$3n$ ($n = 64$ or 128) by repeating step 2 three times to extract the tweakey cells $TK_{19}[0:7], TK_{20}[0:7], TK_{21}[0:7]$. The details of the tweakey schedule can be found in Appendix D in the full version of this paper [13]. Moreover, as in the previous attack on 18-round SKINNY-$n$-$n$ ($n = 64$ or 128), each structure can generate $2^{7\times s} \times 2^{7\times s-1} = 2^{14\times s-1}$ with $2^{7\times s}$ chosen plaintexts. Then, we take $2^m$ structures to generate $2^{m+14\times s-1}$ pairs using $2^{m+7\times s}$ chosen plaintexts.

**Attack Complexity.** The 54 tweakey cells that are involved in the analysis rounds have only $2^{45\times s}$ possible values. The probability that a wrong tweakey is not discarded with one pair is $1 - 2^{27\times s-45\times s} = 1 - 2^{-18\times s}$. Hence, after processing all the $2^{m+14\times s-1}$ pairs, we have $2^{45\times s}(1 - 2^{-18\times s})^{2^{m+14\times s-1}} \approx 2^{45\times s} \times (e^{-1})^{2^{m+14\times s-1-18\times s}} \approx 2^{45\times s} \times 2^{-1.4\times 2^{m-4\times s-1}}$ remaining candidates for $45 \times s$-bit of the tweakey. In order to optimize the time complexity of the

**Table 5.** Time complexity of the different steps of the attack on 18-round SKINNY-64-64 and SKINNY-128-128, where NT denotes the number of tweakeys to be excluded.

| Step | Time Complexity (in 18-round encryptions) | NT | $s = 4, m = 19.52$ | $s = 8, m = 36.42$ |
|---|---|---|---|---|
| 1 | $2^{m+7\times s}$ | - | $2^{47.52}$ | $2^{92.42}$ |
| 4(a) | $2^{m+10\times s-1} \times \frac{1}{16\times 18} \approx 2^{m+10\times s-9.17}$ | 1 | $2^{50.35}$ | $2^{107.25}$ |
| 4(b) | $2^{m+10\times s-1} \times \frac{2}{16\times 18} \approx 2^{m+10\times s-8.17}$ | 1 | $2^{51.35}$ | $2^{108.25}$ |
| 4(c) | $2^{m+10\times s-1} \times \frac{1}{16\times 18} \approx 2^{m+10\times s-9.17}$ | $2^s$ | $2^{50.35}$ | $2^{107.25}$ |
| 5(a) | $2^{m+10\times s-1} \times 2^s \times \frac{1}{16\times 18} \approx 2^{m+11\times s-9.17}$ | $2^s$ | $2^{54.35}$ | $2^{115.25}$ |
| 5(b) | $2^{m+10\times s-1} \times 2^s \times \frac{1}{16\times 18} \approx 2^{m+11\times s-9.17}$ | $2^s$ | $2^{54.35}$ | $2^{115.25}$ |
| 6(a) | $2^{m+10\times s-1} \times 2^s \times \frac{1}{16\times 18} \approx 2^{m+11\times s-9.17}$ | 1 | $2^{54.35}$ | $2^{115.25}$ |
| 7(a) | $2^{m+10\times s-1} \times \frac{1}{16\times 18} \approx 2^{m+10\times s-9.17}$ | $2^{-s}$ | $2^{50.35}$ | $2^{107.25}$ |
| 7(b) | $2^{m+10\times s-1} \times 2^{-s} \times \frac{1}{16\times 18} \approx 2^{m+9\times s-9.17}$ | $2^{-s}$ | $2^{46.35}$ | $2^{99.25}$ |
| 8(a) | $2^{m+10\times s-1} \times 2^{-s} \times \frac{1}{16\times 18} \approx 2^{m+9\times s-9.17}$ | 1 | $2^{46.35}$ | $2^{99.25}$ |
| 9(a) | $2^{m+10\times s-1} \times \frac{1}{16\times 18} \approx 2^{m+10\times s-9.17}$ | 1 | $2^{50.35}$ | $2^{107.25}$ |
| 10(a) | $2^{m+10\times s-1} \times \frac{1}{16\times 18} \approx 2^{m+10\times s-9.17}$ | $2^{-s}$ [a] | $2^{50.35}$ | $2^{107.25}$ |

[a] After this step, we have $2^{-s}$ tweakeys to be excluded for each message pair, i.e., we exclude 1 tweakey after processing $2^s$ pairs.

attack, we choose $m = 19.84$ (resp. $m = 36.22$) in case of SKINNY-64-192 (resp. SKINNY-128-384). Consequently, we have $2^{170}$ (resp. $2^{347}$) remaining tweakey candidates for the 180-bit (resp. 360-bit) of the tweakey. Therefore, the tweakey can be recovered by exhaustively searching the $2^{170}$ (resp. $2^{347}$) remaining tweakey candidates with $2^{12}$ (resp. $2^{24}$) for the other tweakey bits, that are not involved in the attack, using 3 (calculated from the unicity distance) plaintext/ciphertext pairs. Therefore, the total time complexity of the attack is $3 \times 2^{170} \times 2^{12} + 2^{183.97} = 2^{184.79}$ (resp. $3 \times 2^{347} \times 2^{24} + 2^{372.35} = 2^{373.48}$) encryptions in case of SKINNY-64-192 (resp. SKINNY-128-384). The data complexity of the attack is $2^{19.84+28=47.84}$ (resp. $2^{36.22+56=92.22}$) chosen plaintexts in case of SKINNY-64-192 (resp. SKINNY-128-384). The memory complexity of the attack is $2^{74.84}$ (resp. $2^{147.22}$) in case of SKINNY-64-64 (resp. SKINNY-128-384).

## 7 Conclusion

In this work, we presented impossible differential attacks against reduced-round versions of all the 6 SKINNY's variants. All of these attacks use the same impossible differential distinguisher that covers 11-round. We extended this 11-round distinguisher by 7, 9 and 11 rounds to attack 18, 20 and 22 rounds of SKINNY-$n$-$n$, SKINNY-$n$-$2n$ and SKINNY-$n$-$3n$ ($n = 64$ or 128), respectively, exploiting the properties of the MixColumns operation, the simple tweakey schedule and the fact that the tweakey is only added to the first two rows of the state. The presented attacks are currently the best known ones on all the variants of SKINNY in the single-tweakey model.

## References

1. Ankele, R., Banik, S., Chakraborti, A., List, E., Mendel, F., Sim, S. M., Wang, G.: Related-key impossible-differential attack on reduced-round SKINNY. Cryptology ePrint Archive, Report 2016/1127 (2016). http://eprint.iacr.org/2016/1127
2. Beierle, C., Jean, J., Klbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., Sim, S.M.: Skinny family of block ciphers: cryptanalysis competition (2016)
3. Beierle, C., Jean, J., Kölbl, S., Leander, G., Moradi, A., Peyrin, T., Sasaki, Y., Sasdrich, P., Sim, S.M.: The SKINNY family of block ciphers and its low-latency variant MANTIS. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. LNCS, vol. 9815, pp. 123–153. Springer, Heidelberg (2016). doi:10.1007/978-3-662-53008-5_5
4. Biham, E., Biryukov, A., Shamir, A.: Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 12–23. Springer, Heidelberg (1999). doi:10.1007/3-540-48910-X_2
5. Bilgin, B., Gierlichs, B., Nikova, S., Nikov, V., Rijmen, V.: A more efficient AES threshold implementation. In: Pointcheval, D., Vergnaud, D. (eds.) AFRICACRYPT 2014. LNCS, vol. 8469, pp. 267–284. Springer, Cham (2014). doi:10.1007/978-3-319-06734-6_17
6. Daemen, J., Rijmen, V.: The Design of Rijndael. Springer, Heidelberg (2002)

7. Jean, J., Nikolić, I., Peyrin, T.: Tweaks and keys for block ciphers: the TWEAKEY framework. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014. LNCS, vol. 8874, pp. 274–288. Springer, Heidelberg (2014). doi:10.1007/978-3-662-45608-8_15
8. Knudsen, L.: A 128-bit block cipher. Complexity **258**(2), 216 (1998). NIST AES Proposal
9. Liskov, M., Rivest, R.L., Wagner, D.: Tweakable block ciphers. J. Cryptol. **24**(3), 588–613 (2011)
10. Liu, G., Ghosh, M., Song, L.: Security analysis of SKINNY under related-tweakey settings. Cryptology ePrint Archive, Report 2016/1108 (2016). http://eprint.iacr.org/2016/1108
11. Peyrin, T., Seurin, Y.: Counter-in-tweak: authenticated encryption modes for tweakable block ciphers. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. LNCS, vol. 9814, pp. 33–63. Springer, Heidelberg (2016). doi:10.1007/978-3-662-53018-4_2
12. Sadeghi, S., Mohammadi, T., Bagheri, N.: Cryptanalysis of reduced round SKINNY block cipher. Cryptology ePrint Archive, Report 2016/1120 (2016). http://eprint.iacr.org/2016/1120
13. Tolba, M., Abdelkhalek, A., Youssef, A.M.: Impossible differential cryptanalysis of reduced-round skinny. Cryptology ePrint Archive, Report 2016/1115 (2016). http://eprint.iacr.org/2016/1115