Hindawi
Security and Communication Networks
Volume 2020, Article ID 8864102, 8 pages
https://doi.org/10.1155/2020/8864102

WILEY | Hindawi

Research Article

Known-Key Distinguishing and Partial-Collision Attacks on GFN-2 with SP F-Function

Deukjo Hong

Jeonbuk National University, Jeonju, Republic of Korea

Correspondence should be addressed to Deukjo Hong; deukjo.hong@jbnu.ac.kr

Received 23 August 2020; Accepted 3 October 2020; Published 28 October 2020

Academic Editor: Petros Nicopolitidis

We study known-key distinguishing and partial-collision attacks on GFN-2 structures with various block lengths in this paper. For 4-branch GFN-2, we present 15-round known-key distinguishing attack and 11-round partial-collision attack which improve previous results. We also present 17-round known-key distinguishing attack on 6-branch GFN-2 and 27-round known-key distinguishing attack on 8-branch GFN-2 and show that several partial-collision attacks are derived from them. Additionally, some attacks are valid under special conditions for the F-function.

1. Introduction

The notion of known-key attack was introduced by Knudsen and Rijmen in 2007 [1]. It uses a known-key distinguisher which holds with much higher probability than that under the uniform distribution. In 2011, Sasaki and Yasuda used the rebound technique [2] to construct known-key distinguishers for the Feistel network whose F-function consists of cryptographically strong S-boxes and an MDS matrix and showed that those distinguishers are converted into partial-collision attacks on hash modes [3]. Later, their results have been applied to variants of the Feistel network [4–6].

Feistel network is the encryption structure of well-known block ciphers such as DES [7], SEED [8], and Camellia [9]. It has been researched for secure and efficient block cipher design. In [4], Kang et al. presented known-key attacks on three types of generalized Feistel network (GFN) proposed by Nyberg [10]. Particularly, Type-II GFN (GFN-2) is well-balanced like Feistel network and suitable for lightweight designs because the iteration of the relatively small F-function makes a large-block-length block cipher. So, it has been researched as an alternative of Feistel network, more than other types of GFN. It is often considered as one of design candidates in developing new block ciphers. In practice, the encryption structure of CLEFIA [11] is GFN-2, and HIGHT [12] adopted a slight variant of GFN-2. For this reason, it is important and useful to study and analyze the security of GFN-2.

We define GFN-2 with the parameters t, a, and b. t is the number of branches, a is the number of S-boxes which the F-function consists of, and b is the length of input and output of the bijective S-box. In this paper, the byte length and the word length are defined as b bits and ab bits. The block length of GFN-2 with the parameters t, a, and b is abt bits. We restrict (a, b) to (4, 4), (4, 8), (8, 4), and (8, 8) and t to 4, 6, and 8, which are mainly used and considered in block cipher designs.

In [4], Kang et al. analyzed only $t = 4$ cases of GFNs and assumed that the last-round function has no shuffle operation. They presented a 13-round known-key distinguishing attack on GFN-2 and 9-round 1-word and 2-word partial-collision attacks on Matyas-Meyer-Oseas and Miyaguchi-Preneel hash modes of GFN-2. In this paper, we improve the results for GFN-2 in [4] and also present known-key distinguishing and partial-collision attacks for the cases of $t = 6$ and $t = 8$. Our results are summarized as follows:

(i) For 4-branch GFN-2, we find a new 5-round inbound structure and make a 15-round known-key distinguishing attack. Assuming the last round has no shuffle operation, we show that a 11-round 3-

word partial-collision attack is possible and that when $a = 8$, 15-round 1-word partial-collision attack is possible. Assuming the last round has the shuffle operation, we show that a 10-round 3-word partial-collision attack is possible and that when $a = 8$, 14-round 1-word partial-collision attack is possible.

(ii) For 6-branch GFN-2, we find a 7-round inbound structure and make a 17-round known-key distinguishing attack. When $a = 8$, we show that a 19-round known-key distinguishing attack, a 17-round 2-word partial-collision attack without the last shuffle operation, and a 16-round 2-word partial-collision attack with the last shuffle operation are possible.

(iii) For 8-branch GFN-2, we find a 11-round inbound structure and make a 27-round known-key distinguishing attack which is extended to 29 rounds when $a = 8$. We show that a 21-round 5-word partial-collision attack without the last-round shuffle operation and a 20-round 5-word partial-collision attack with the last-round shuffle operation are possible and that a 21-round 2-word partial-collision attack with the last-round shuffle operation is possible when $(a, b) \neq (4, 8)$.

Considering the wide applicability of GFN-2 as a structure of the cryptographic algorithm, our attacks are useful and helpful in designing a new block cipher or hash function based on GFN-2. The remainder of this paper is organized as follows: Section 2 gives a brief description of GFN-2 structure and Matyas-Meyer-Oseas and Miyaguchi-Preneel mode and explains the inbound structure of F-function. Section 3 provides a general explanation of how to construct an inbound structure for GFN-2. From Section 4 to Section 6, we propose inbound structures, known-key distinguishers, and partial-collision attacks on GFN-2 for $t = 4$, 6, and 8. Finally, Section 7 concludes our work.

2. Preliminaries

2.1. Type-2 Generalized Feistel Network. Let the S-box S: $\{0, 1\}^b \longrightarrow \{0, 1\}^b$ be a nonlinear permutation on $\{0, 1\}^b$. The notation $Y = S(X)$ means that the output of the S-box is $Y \in \{0, 1\}^b$ on the input $X \in \{0, 1\}^b$. Let the linear function P: $(\{0, 1\}^b)^a \longrightarrow (\{0, 1\}^b)^a$ be the multiplication by $a \times a$ MDS matrix over GF (2^b). The notation $Y = (Y[0], Y[1], \ldots, Y[a-1]) = P(X) = P(X[0], X[1], \ldots, X[a-1])$ means that the output vector of P is $Y = (Y[0], Y[1], \ldots, Y[a-1])$ on the input vector $X = (X[0], X[1], \ldots, X[a-1])$. For the S-box S and the linear function P, we define F: $\{0, 1\}^{ab} \times \{0, 1\}^{ab}$ as follows: for an input $X = (X[0], X[1], \ldots, X[a-1]) \in (\{0, 1\}^b)^a$ and a subkey $RK = (RK[0], RK[1], \ldots, RK[a-1]) \in (\{0, 1\}^b)^a$, $Y = F(X, RK) = P(S(X[0] + RK[0]), S(X[1] + RK[1]))$, where + is the XOR (eXclusive OR) operation. Figure 1 depicts the example of F-function with $a = 4$.

Let $t \geq 4$ be an even integer and r be a positive integer. For r-round t-branch GFN-2, we define all subkeys $RK_{i,j}$ generated from a key K as $RK_{i,j} = (RK_{i,j}[0], RK_{i,j}[1], \ldots, RK_{i,j}[a-1]) \in (\{0, 1\}^b)^a$ for $0 \leq i < r$ and $0 \leq j < t/2$.

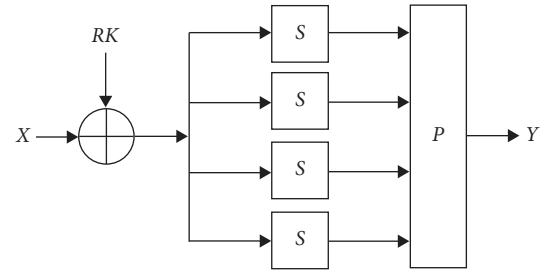

Figure 1: Structure of F-function with $a = 4$.

We define the shuffle operation σ as $\sigma = (\sigma(0), \sigma(1), \ldots, \sigma(t-1)) = (t-1, 0, \ldots, t-2)$. Then, we can give the following pseudocode which describes how the r-round t-branch GFN-2 encrypts a plaintext block $X_0 = (X_{0,0}, X_{0,1}, \ldots, X_{0,t-1}) \in (\{0, 1\}^{ab})^t$ to $X_r = (X_{r,0}, X_{r,1}, \ldots, X_{r,t-1}) \in (\{0, 1\}^{ab})^t$:

(i) **for** $i = 0, 1, \ldots, r-1$ **do**:
(ii) **for** $j = 0, 1, \ldots, t-1$ **do**:
(iii) **if** j is even:
(iv) $Y_j = X_{i,j}$
(v) **else**:
(vi) $Y_j = X_{i,j} + F(X_{i,j-1}, RK_{i,(j-1)/2})$
(vii) **for** $j = 0, 1, \ldots, t-1$ **do**:
(viii) $X_{i+1,\sigma(j)} = Y_j$

The index i in the above pseudocode means the round order. Figure 2 depicts the i-th round function of GFN-2 with $t = 8$. Throughout this paper, we assume that the key K and the subkey $RK_{i,j}$'s are known and fixed. Since subkey-XORing operations are not important in the description of our work, we omit the notation and explanation about subkeys for simplicity. For example, we replace $F(X_{i,j-1}, RK_{i,(j-1)/2})$ with $F(X_{i,j-1})$.

2.2. Inbound Structure of F-Function. A difference is the XOR between two values at the same position, and a differential trail is a set of all difference transitions in a block cipher. An inbound structure is a core part in rebound attack techniques [2] and is a set of all pairs satisfying a differential trail for a part of a block cipher. In order to give an easy explanation about inbound structure of F-function (ISF), we need to use the following notations of word difference forms:

(i) 0: every byte in the word has the zero difference.

(ii) Δ_1: one byte has a nonzero difference and the other bytes in the word have zero differences.

(iii) $\Delta_{P(1)}$: the word has difference forms which are the output of P on the input Δ_1. That is, $P(\Delta_1) = \Delta_{P(1)}$.

We assume that all subkeys are known and fixed and that the number of zero entries is almost equal to that of nonzero entries in the difference distribution table (DDT) of the S-box. We set the input and output difference forms of the F-function to $\Delta_{P(1)}$ and Δ_1, respectively. Then, for all possible

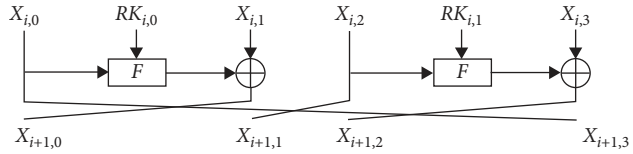

Figure 2: Structure of F-function with $a = 4$.

differences with the form of $(\Delta_{P(1)}, \Delta_1) \in \{0, 1\}^{ab} \times \{0, 1\}^{ab}$, every S-box in the F-function meets nonzero input and output differences. For any choice of nonzero difference pair $(\alpha, \beta) \in \{0, 1\}^b \times \{0, 1\}^b$, we call it valid if there exists any input pair whose input difference is α and the corresponding S-box output difference is β. By the assumption of DDT, the ratio of valid input and output difference pairs is around 0.5. On average, for a valid input-output difference pair (α, β), the S-box has a single input pair $(x_1, x_2) \in \{0, 1\}^b \times \{0, 1\}^b$ satisfying $x_1 + x_2 = \alpha$ and $S(x_1) + S(x_2) = \beta$. For any fixed form $\Delta_{P(1)}$, the number of input differences of the F-function satisfying $\Delta_{P(1)}$ is $2^b - 1$. For any fixed form $\Delta_{P(1)}$, the number of input differences of the F-function satisfying Δ_1 is $2^b - 1$. Therefore, on average, the inbound structure of the F-function (ISF) with $(\Delta_{P(1)}, \Delta_1)$ contains $(2^b - 1)^2 \times 2^{-a} \times 2^a = (2^b - 1)^2 \cong 2^{2b}$.

We take a look at the example of ISF with $a = 4$ in Figure 3. Let the input differences of the four S-boxes be α_0, α_1, α_2, and α_3, and let the corresponding output differences be β_0, β_1, β_2, and β_3. Let $x_{0,0}$, $x_{0,1}$, $x_{1,0}$, $x_{1,1}$, $x_{2,0}$, $x_{2,1}$, $x_{3,0}$, and $x_{3,1}$ be the inputs of the S-box satisfying

$$\begin{aligned} x_{i,0} + x_{i,1} &= \alpha_i, \\ S\left(x_{i,0}\right) + S\left(x_{i,1}\right) &= \beta_i, \quad \text{for } i \in \{0, 1, 2, 3\}. \end{aligned} \tag{1}$$

Then, all the possible input pairs of the F-function are $\{(x_{0,0}, x_{1,0}, x_{2,0}, x_{3,0}), (x_{0,1}, x_{1,1}, x_{2,1}, x_{3,1})\}$, $\{(x_{0,1}, x_{1,0}, x_{2,0}, x_{3,0}), (x_{0,0}, x_{1,1}, x_{2,1}, x_{3,1})\}$, $\{(x_{0,0}, x_{1,1}, x_{2,0}, x_{3,0}), (x_{0,1}, x_{1,0}, x_{2,1}, x_{3,1})\}$, ... That is, the number of possible input pairs of the F-function is 2^4. Therefore, the ISF contains about $(2^b - 1)^2$ pairs because the F-function has about $(2^b - 1)^2 \times 2^{-4}$ possible input-output difference pairs with the form $(\Delta_{P(1)}, \Delta_1)$.

We assume that DDT of the S-box is given in advance and that DDT contains all possible input pairs for each input and output differences. Then, the complexity of the phase checking the validity of an input-output difference pair for the S-box is dominant in the computational complexity required for constructing the ISF. It is about $a \times 2^{2b}$ table lookups $\cong 2^{2b}$ F-function evaluations because the F-function consists of a S-boxes.

2.3. Matyas-Meyer-Oseas and Miyaguchi-Preneel Modes. Matyas-Meyer-Oseas (MMO) and Miyaguchi-Preneel (MP) modes belong to 12 secure PGV hash modes, [15] which invoke a single call of the underlying block cipher to build a compression for a Merkle-Damgård hash function. Note that a compression function takes a message block and an input chaining variable value to produce an output chaining variable value. In both of two hash modes, the input chaining variable which cannot be controlled by anyone becomes the key of the block cipher, the message block which can be controlled by anyone becomes the plaintext block of the block cipher, and the output chaining variable is produced by XORing the ciphertext block with the plaintext block and the key. See Figure 4. Throughout this paper, we assume the hash mode of GFN-2 is MMO or MP whenever we explain partial-collision attacks.

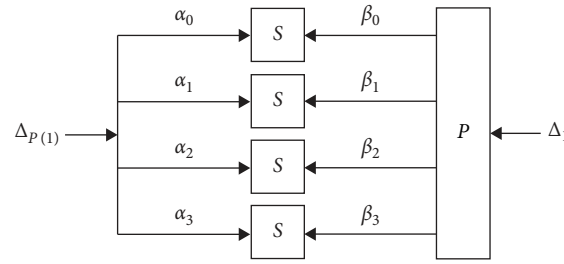

Figure 3: Differences in the inbound structure of F-function with $a = 4$.

3. Inbound Structure of GFN-2

We explore the inbound structures of GFN-2 (ISG2) which minimize nonzero difference words with the form Δ_1. Such ISG2s have relatively long difference propagation in forward and backward directions and best attacks on hash modes. We suggest a general methodology to construct differential trails suitable for good ISG2s as follows:

(1) Set the round number R of ISG2 to an intended positive integer.

(2) Select the number of ISFs and randomly choose the application positions of ISFs. For each chosen position, set the input and output differences of the F-function to $\Delta_{P(1)}$ and Δ_1, respectively.

(3) Use only the difference forms 0, Δ_1, and $\Delta_{P(1)}$ to propagate and adjust the differences from ISFs in forward and backward directions such that nonzero differences are minimized.

(4) Check whether the input and output differences of ISG2 have the minimum number of nonzero word differences with the form Δ_1. If it is, return the differential trail; otherwise, go to Step (2).

We assume that the position of nonzero byte in Δ_1 is the same as that in $P^{-1}(\Delta_{P(1)})$ and that all subkeys are known and fixed. Let "?" be an unknown difference. We use the notation "0," "Δ_1," "$\Delta_{P(1)}$," and "?" to represent the difference forms. We make them correspond to binary codes 00_2, 01_2, 10_2, and 11_2. Then, the difference form of two consecutive words can be represented in hexadecimal digits like Table 1. For example, $(\Delta_1, \Delta_{P(1)})$ and $(\Delta_{P(1)}, \Delta_1)$ are 0×6 and 0×9, respectively.

4. Attacks on 4-Branch GFN-2

4.1. 5-Round Inbound Structure. We make the 5-round inbound structure satisfying the differential trail in Figure 5. It is represented as a hexadecimal vector (0×40, 0×81, 0×46,

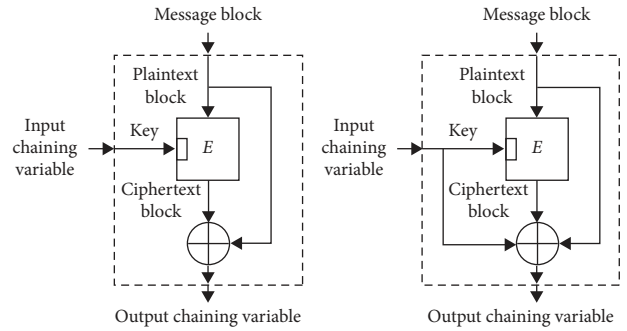

Figure 4: Matyas-Meyer-Oseas (left) mode and Miyaguchi-Preneel (right) mode.

Table 1: Hexadecimal representation for two consecutive words.

	0	Δ_1	$\Delta_{P(1)}$	?
0	0x0	0x1	0x2	0x3
Δ_1	0x4	0x5	0x6	0x7
$\Delta_{P(1)}$	0x8	0x9	0xA	0xB
?	0xC	0xD	0xE	0xF

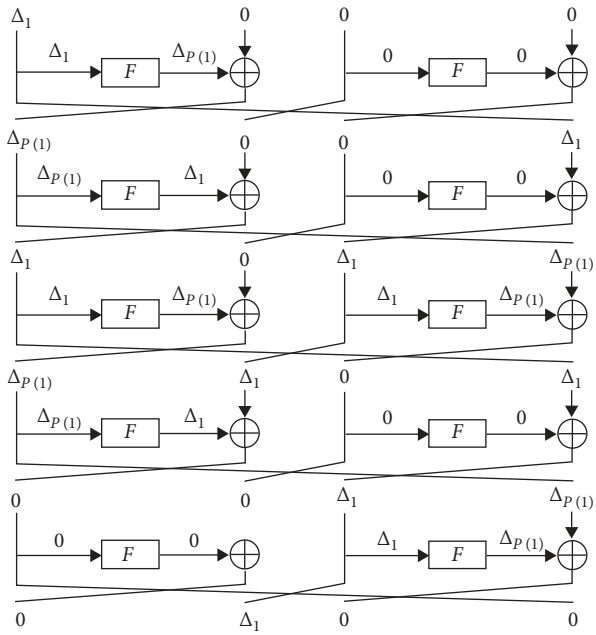

Figure 5: Differential trail for the 5-round inbound structure of 4-branch GFN-2.

0×91, 0×06, 0×10) by Table 1. The input state of ISG2 is $X_0 = (X_{0,0}, X_{0,1}, X_{0,2}, X_{0,3})$ and the output state of the i-th round is $X_{i+1} = (X_{i+1,0}, X_{i+1,1}, X_{i+1,2}, X_{i+1,3})$ for $i \in \{0, 1, 2, 3, 4\}$. Let $\Delta X_{i,j}$ be the difference at $X_{i,j}$ and let $\Delta F(X_{i,j})$ be the difference at $F(X_{i,j})$. We use two ISFs to find pairs contained in the 5-round ISG2 according to the following steps:

(1) Apply the ISF to the F-function taking $X_{1,0}$ as input. Store about 2^{2b} pairs satisfying the input difference with the form $\Delta_{P(1)}$ and the output difference with the form Δ_1 for the F-function, in a table named ISF-1.

(2) Apply the ISF to the F-function taking $X_{3,0}$ as input, independently of Step (1). Store about 2^{2b} pairs satisfying the input difference with the form $\Delta_{P(1)}$ and the output difference with the form Δ_1 for the F-function, in a table named ISF-2.

(3) Choose a random value for $X_{0,2}$ and compute $F(X_{0,2})$. Then, compute $X_{2,0}$ and $F(X_{2,0})$ for all values of $F(X_{1,0})$ in ISF-1.

(4) Choose a random value for $X_{4,0}$ (=$X_{5,3}$) and compute $F(X_{4,0})$. Then, compute $X_{2,2}$ and $F(X_{2,2})$ for all values of $F(X_{3,0})$ in ISF-2.

(5) For $\Delta F(X_{2,0})$ and $\Delta X_{1,0}$ from a pair $(x_1, x_2) \in$ ISF-1 and for $\Delta X_{3,0}$ and $\Delta F(X_{2,2})$ from a pair $(y_1, y_2) \in$ ISF-2, combine the pairs to $\{(x_1, y_1), (x_2, y_2)\}$ if $\Delta F(X_{2,0}) = \Delta X_{3,0}$ and $\Delta X_{1,0} = \Delta F(X_{2,2})$. For all the pairs in ISF-1 and all the pairs in ISF-2, store the combined pairs in a table named ISF-(1, 2). On average, ISF-(1, 2) contains 2^{2b} (combined) pairs.

(6) For all pairs in ISF-(1, 2), compute $F(X_{1,2})$ and $F(X_{0,0})$, and discard the pairs where the difference at $F(X_{0,0})$ is not equal to the difference at $X_{1,0}$. On average, 2^b pairs survive.

(7) For all surviving pairs, compute $F(X_{3,2})$ and $F(X_{4,2})$, and discard the pairs where the difference at $F(X_{4,2})$ is not equal to the difference at $X_{3,0}$. On average, 1 pair survives.

(8) For all surviving pairs, compute $X_{0,1}$, $X_{0,3}$, $X_{5,0}$, and $X_{5,2}$, and store the resulting pairs in ISG2.

For fixed values of $X_{0,1}$ and $X_{5,3}$, the above 5-round ISG2 has one pair on average. The computational complexity required for constructing a 5-round ISG2 is estimated 9×2^{2b} F-function evaluations. We denote it by $T \cong 9 \cdot 2^{2b} F$. This estimation is based on the following:

(i) The construction of ISFs for Steps (1) and (2) requires $2^{2b}F$ because essentially, a single set of ISF can be applied to two positions.

(ii) The complexity of Step (3) is $2^{2b+1}F$ because $F(X_{2,0})$ is computed for 2^{2b+1} times.

(iii) The complexity of Step (4) is $2^{2b+1}F$ because $F(X_{2,2})$ is computed for 2^{2b+1} times.

(iv) The complexity of Step (6) is $2^{2b+2}F$ because $F(X_{1,2})$ is computed for 2^{2b+1} times and $F(X_{0,0})$ is computed for 2^{2b+1} times.

(v) The complexity of Step (7) is $2^{b+2}F$ (=$2 \times 2 \times 2^b F$) because $F(X_{3,2})$ is computed for 2^{b+1} times and $F(X_{4,2})$ is computed for 2^{b+1} times.

So, if we choose N random values of $(X_{0,1}, X_{5,3})$, the 5-round ISG2 contains N pairs and the corresponding complexity is NT.

4.2. Known-Key Distinguisher. We can get a differential trail in Table 2 by propagating differences from the 5-round ISG2 in forward and backward directions. $\Delta X_i = (\Delta X_{i,0}, \ldots, \Delta X_{i,3})$ is the representation of the difference of the state. ISG2

Table 2: Difference propagation from the 5-round inbound structure of 4-branch GFN-2.

i	$\Delta X_{i,0}$	$\Delta X_{i,1}$	$\Delta X_{i,2}$	$\Delta X_{i,3}$
−5	?	?	$\Delta_{P(1)}$	?
−4	Δ_1	$\Delta_{P(1)}$	?	?
−3	$\Delta_{P(1)}$	?	0	Δ_1
−2	0	0	Δ_1	$\Delta_{P(1)}$
−1	0	Δ_1	0	0
0	Δ_1	0	0	0
1	$\Delta_{P(1)}$	0	0	Δ_1
2	Δ_1	0	Δ_1	$\Delta_{P(1)}$
3	$\Delta_{P(1)}$	0	0	Δ_1
4	0	0	Δ_1	$\Delta_{P(1)}$
5	0	Δ_1	0	0
+1	Δ_1	0	0	0
+2	$\Delta_{P(1)}$	0	0	Δ_1
+3	?	0	Δ_1	$\Delta_{P(1)}$
+4	?	Δ_1	$\Delta_{P(1)}$	?
+5	?	$\Delta_{P(1)}$	?	?

covers from ΔX_0 to ΔX_5, the backward propagation covers from ΔX_{-1} to ΔX_{-5}, and the forward propagation covers from ΔX_{+1} to ΔX_{+5}. The rebound attack framework calls this propagation, Outbound Phase [2]. In this phase, the transition between the input and output difference forms under the F-function is determined by the rule in Table 3.

The differential trail in Table 2 is represented as 0xFB ⟶ 0xEF by hexadecimal digits. In Table 2, the difference form at $X_{i,j}$ is denoted by $\Delta X_{i,j}$. In the case of ideal cipher with the block length of abt bits, we explain how to find at least one pair satisfying 0xFB ⟶ 0xEF. Firstly, we make a set of 2^b abt-bit values such that all possible byte values appear at the nonzero byte difference, which is indicated by the difference form $\Delta_{P(1)}$, and a randomly chosen constant value is at the zero byte differences. After applying the linear function P to the third words of the elements in the set, we get about 2^{2b-1} pairs with the difference form (?, ?, $\Delta_{P(1)}$, ?). Then, the output difference form is (?, $\Delta_{P(1)}$, ?, ?) with the probability $2^{-(a-1)b}$, and we get $2^{(-a+3)b-1} = 2^{2b-1} \times 2^{-(a-1)b}$ pairs satisfying 0×FB ⟶ 0×EF. Since $a = 4$ or $a = 8$ in the block cipher designs, $(-a+3)b-1$ is a negative integer. Therefore, we expect a pair satisfying 0xFB ⟶ 0xEF by repeating this work $2^{(a-3)b+1} = 1/2^{(-a+3)-1}$ times, and the complexity is $2^{(a-2)b+1} = 2^b \times 2^{(a-3)b+1}$.

In the case of 4-branch GFN-2, we can get one pair satisfying 0xFB ⟶ 0xEF with $9 \times 2^{2b} F = 9 \times 2^{2b}/30$ because a pair contained in the 5-round ISG2 satisfies 0xFB ⟶ 0xEF, the complexity required in the computation of the outbound phase is negligible, and one evaluation of the 15-round 4-branch GFN-2 requires 30 evaluations of the F-function. When $a = 4$ or $a = 8$, the complexity in the case of GFN-2 is lower than that of the ideal cipher and so, 0xFB ⟶ 0xEF can be used as a valid 15-round known-key distinguisher. By the way, the attack advantage in the case of $a = 4$ is much smaller than that of $a = 8$.

The summary of the attack complexity can be seen in Table 4. The validity of the distinguishing attack has nothing to do with the existence of the shuffle operation in the last round, but we just write the distinguishing attack result in the case that the shuffle operation exists in the last round.

Table 3: Transition between the input and output difference forms under the F-function in the outbound phase.

Input difference	Output difference
0	0
Δ_1	$\Delta_{P(1)}$
$\Delta_{P(1)}$	?
?	?

4.3. Partial-Collision Attacks. The partial-collision attacks derived from Table 2 are summarized in Table 4. The "L" column in Table 4 means the existence of the last shuffle operation; if the last shuffle operation exists, its entry is "Y"; otherwise, its entry is "N." The "R" column means the number of attacked rounds. The "KKD" column means the known-key distinguisher used in each attack; the entry is written with the forms of input difference and output difference. The "w" column means the number of words colliding at the output chaining variable in the partial-collision attack. For the first attack in Table 4, its entry is written as "–" because it is a just distinguishing attack. The "Comp." column means the complexity required for the known-key distinguishing attack or partial-collision attack on 4-branch GFN-2, and the "Generic" column means the complexity required for the known-key distinguishing attack on the ideal cipher with abt-bit block or the birthday attack on a random function with abt-bit output length. Finally, the "(a, b)" column means the value of (a, b) which makes the attack valid; its entry is written as "all" if the attack is valid for all values of (a, b); its entry is written as "$(8, *)$" if the attack is valid only for $a = 8$.

The second attack in Table 4 uses known-key distinguishers (?, ?, $\Delta_{P(1)}$, ?) ⟶ (?, Δ_1, $\Delta_{P(1)}$, ?) or (Δ_1, $\Delta_{P(1)}$, ?, ?) ⟶ (?, $\Delta_{P(1)}$, ?, ?), and we expect a 1-word partial collision by trying 2^b pairs in ISG2. Since it covers 14 rounds, the complexity is estimated as $2^b \times (9 \times 2^{2b})/28 = 2^{3b-1.63}$. This attack is valid only for $a = 8$ because the complexity is lower than $2^{ab/2}$ when $a = 8$.

The third attack in Table 4 uses known-key distinguishers ($\Delta_{P(1)}$, ?, 0, Δ_1) ⟶ ($\Delta_{P(1)}$, 0, 0, Δ_1) or (0, 0, Δ_1, $\Delta_{P(1)}$) ⟶ (?, 0, Δ_1, $\Delta_{P(1)}$), and we expect a 3-word partial collision by 2^{2b} pairs in ISG2. Since it covers 10 rounds, the complexity is estimated as $2^{2b} \times (9 \times 2^{2b})/20 = 2^{3b-1.15}$. This attack is valid for all values $(a, b) \in \{(4, 4), (4, 8), (8, 4), (8, 8)\}$.

The fourth and fifth attacks in Table 4 use known-key distinguishers (?, ?, $\Delta_{P(1)}$, ?) ⟶ (?, ?, $\Delta_{P(1)}$, ?) and ($\Delta_{P(1)}$, ?, 0, Δ_1) ⟶ ($\Delta_{P(1)}$, 0, 0, Δ_1), respectively, under the assumption that the last round has no shuffle operation. The complexity and validity of them are understood by the similar way to the second and third attacks.

5. Attacks on 6-Branch GFN-2

We make a 7-round ISG2 for 6-branch GFN-2, and the corresponding differential trail is represented as a hexadecimal vector (0×400, 0×801, 0×406, 0×811, 0×446, 0×991, 0×606, 0×011). We use four ISFs to find the pairs for the 7-round ISG2 according to the following steps:

Table 4: Known-key distinguishing and w-word partial-collision attacks on 4-branch GFN-2.

L	R	KKD	w	Comp.	Generic	(a, b)
Y	15	(0xFB, 0xEF)	—	$2^{2b-1.73}$	$2^{(a-2)b+1}$	All
	14	(0xFB, 0xDB), (0x6F, 0xEF)	1	$2^{3b-1.63}$	$2^{ab/2}$	$(8, *)$
	10	(0xB1, 0x81), (0x06, 0xC6)	3	$2^{4b-1.15}$	$2^{3ab/2}$	All
N	15	(0xFB, 0xB)	1	$2^{3b-1.73}$	$2^{ab/2}$	$(8, *)$
	11	(0xB1, 0x81)	3	$2^{4b-1.29}$	$2^{3ab/2}$	All

(1) Apply ISFs to the F-functions taking $X_{1,0}$, $X_{3,0}$, $X_{5,0}$, and $X_{5,2}$ as inputs. Call them ISF-1, ISF-2, ISF-3, and ISF-4, respectively.

(2) Choose a random value for $X_{0,2}$ to compute $F(X_{0,2})$, and compute $F(X_{2,0})$ for all values of $F(X_{1,0})$ in ISF-1. Then, for $\Delta F(X_{2,0})$ associated to a pair $(x_1, x_2) \in$ ISF-1 and for $\Delta X_{3,0}$ from a pair $(y_1, y_2) \in$ ISF-2, combine the pairs to $\{(x_1, y_1), (x_2, y_2)\}$ if $\Delta F(X_{2,0}) = \Delta X_{3,0}$. For all the pairs in ISF-1 and all the pairs in ISF-2, store the combined pairs in a table named ISF-(1, 2). On average, ISF-(1, 2) contains $2^{3b} = 2^{2b} \times 2^{2b} \times 2^{-b}$ pairs.

(3) Choose a random value for $X_{7,1}$ $(=X_{6,2})$ to compute $F(X_{6,2})$, and compute $F(X_{4,4})$ for all values of $F(X_{5,2})$ in ISF-4. Then, for $\Delta F(X_{4,4})$ associated to a pair $(z_1, z_2) \in$ ISF-4 and for $\Delta X_{3,0}$ from a pair $\{(x_1, y_1), (x_2, y_2)\} \in$ ISF-(1, 2), combine the pairs to $\{(x_1, y_1, z_1), (x_2, y_2, z_2)\} \in$ if $\Delta F(X_{4,4}) = \Delta X_{3,0}$. For all pairs in ISF-4 and all pairs in ISF-(1, 2), store the combined pairs in a table named ISF-(1, 2, 4). On average, ISF-(1, 2, 4) contains $2^{2b} \times 2^{3b} \times 2^{-b} = 2^{4b}$ pairs.

(4) For each pair in ISF-(1, 2, 4), discard it if $\Delta X_{3,5} \neq \Delta X_{4,4}$. On average, ISF-(1, 2, 4) contains $2^{4b} \times 2^{-b} = 2^{3b}$ pairs after this filtering.

(5) For each pair in ISF-(1, 2, 4), compute $X_{3,4}$, $X_{2,4}$, and $X_{4,2}$, and discard the pair if $\Delta X_{2,4} \neq \Delta X_{4,2}$. On average, ISF-(1, 2, 4) contains $2^{3b} \times 2^{-b} = 2^{2b}$ after this filtering.

(6) For all pairs $\{(x_1, y_1, z_1), (x_2, y_2, z_2)\}$ in ISF-(1, 2, 4) and all pairs (v_1, v_2) in ISF-3, compute $X_{3,2}$ and $X_{4,0}$. Then, combine the pairs to $\{(x_1, y_1, z_1, v_1), (x_2, y_2, z_2, v_2)\}$ if $\Delta X_{4,0} = \Delta F(X_{3,0})$, and store the combined pairs in a table ISG2. On average, ISG2 contains $2^{2b} \times 2^{2b} \times 2^{-b} = 2^{3b}$ pairs.

(7) For each pair in ISG2, compute $X_{6,0}$ and $F(X_{6,0})$ and then discard the pair if $\Delta X_{6,0} = 0$ or $\Delta F(X_{6,0}) \neq \Delta X_{5,2}$. On average, ISG2 contains $2^{3b} \times 2^{-b} = 2^{2b}$ pairs after this filtering.

(8) For each pair in ISG2, compute $F(X_{5,4})$ and $F(X_{6,4})$, and then discard the pair if $\Delta F(X_{6,4}) \neq \Delta X_{5,0}$. On average, ISG2 contains $2^{2b} \times 2^{-b} = 2^{b}$ pairs after this filtering.

(9) For each surviving pair in ISG2, compute the remaining parts including $F(X_{0,0})$, and discard the pair if $\Delta F(X_{0,0}) \neq \Delta X_{1,0}$. On average, ISG2 contains $2^{b} \times 2^{-b} = 1$ pair after this filtering.

That is, for a fixed $X_{0,2}$ and $X_{7,1}$, we can find a pair for the 7-round ISG2 of 6-branch GFN-2. The complexity of the above procedure is about $2^{4b+1}F$. It is based on the following and we can see that the complexity of Step (6) is dominant:

(i) The construction of ISFs for Step (1) requires $2^{2b}F$.

(ii) The complexity of Step (2) is $2^{2b+1}F$ because $F(X_{2,0})$ is computed for 2^{2b+1} times.

(iii) The complexity of Step (3) is $2^{2b+1}F$ because $F(X_{4,4})$ is computed for 2^{2b+1} times.

(iv) The complexity of Step (5) is $3 \times 2^{3b+1}F$ because $X_{3,4} = F^{-1}(X_{2,0} + X_{4,4})$, $X_{2,4} = F^{-1}(X_{1,0} + X_{3,4})$, and $X_{4,2} = F^{-1}(X_{3,4} + X_{5,2})$ are computed for 2^{3b+1} times, where we assume that the evaluation of F^{-1} requires the same complexity as F.

(v) The complexity of Step (6) is $2^{4b+1}F$ because $X_{3,2} = F^{-1}(X_{2,4} + X_{4,2})$ is computed for 2^{2b+1} times and $F^{-1}(X_{3,2} + X_{5,0})$ is computed for 2^{4b+1} times.

(vi) The complexity of Step (7) is $2^{3b+1}F$ because $F(X_{6,0})$ is computed for 2^{3b+1} times.

(vii) The complexity of Step (8) is $2^{2b+2}F$ because $F(X_{5,4})$ and $F(X_{6,4})$ are computed for 2^{2b+1} times.

(viii) The complexity of Steps (4) and (9) is negligible compared to the other steps.

Table 5 summarizes known-key distinguishing and partial-collision attacks on 6-branch GFN-2, based on the 7-round ISG2. The first attack in Table 5 is a 19-round known-key distinguishing attack. The condition that a known-key distinguisher for 6-branch GFN-2 is valid for all values of (a, b) is that the distinguisher has more than two nonzero words in both input and output differences. The 17-round known-key distinguisher 0x6FF $\longrightarrow$ 0xBFD is the longest one which is valid for all values of (a, b). Table 5 shows that the partial-collision attacks on 6-branch GFN-2 are valid only for $a = 8$.

6. Attacks on 8-Branch GFN-2

We make a 11-round ISG2 for 8-branch GFN-2, and the corresponding differential trail is represented as a hexadecimal vector $(0 \times 4000, 0 \times 8001, 0 \times 4006, 0 \times 8011, 0 \times 4046, 0 \times 8191, 0 \times 4606, 0 \times 9011, 0 \times 0046, 0 \times 0190, 0 \times 0600, 0 \times 1000)$. The procedure finding a pair for the 11-round ISG2 of 8-branch GFN-2 is described as follows:

Table 5: Known-key distinguishing and w-word partial-collision attacks on 6-branch GFN-2.

L	R	KKD	w	Comp.	Generic	(a, b)
Y	19	(0xFBF, 0xFFE)	—	$2^{4b-4.83}$	$2^{(a-2)b+1}$	$(8, *)$
	17	(0x6FF, 0xBFD)	—	$2^{4b-4.76}$	$2^{2(a-1)b+1}$	All
	16	(0x6FF, 0x6EC), (0xBF1, 0xBFD)	2	$2^{6b-4.59}$	2^{ab}	$(8, *)$
N	17	(0x6FF, 0x6FF)	2	$2^{6b-4.67}$	2^{ab}	$(8, *)$

Table 6: Known-key distinguishing and w-word partial-collision attacks on 8-branch GFN-2.

L	R	KKD	w	Comp.	Generic	(a, b)
Y	29	(0xFBFF, 0xEFFF)	—	$2^{5b-5.86}$	$2^{(a-2)b+1}$	$(8, *)$
	21	(0x6FFF, 0xDBFF)	—	$2^{5b-5.76}$	$2^{2(a-1)b+1}$	All
	27	(0xF01B, 0xC06F), (0x006F, 0xC1BF)	2	$2^{5b-5.39}$	2^{ab}	$\neg(4, 8)$
	20	(0xF01B, 0xC01B), (0x006F, 0xC06F)	5	$2^{7b-5.32}$	$2^{5ab/2}$	All
N	21	(0xF01B, 0xF01B)	5	$2^{7b-5.39}$	$2^{5ab/2}$	All

(1) Apply six ISFs to the F-functions taking $X_{1,0}$, $X_{3,0}$, $X_{5,0}$, $X_{5,4}$, $X_{7,0}$, and $X_{9,4}$ as inputs. Call them ISF-1, ISF-2, ISF-3, ISF-4, ISF-5, and ISF-6, respectively.

(2) Choose a random value for $X_{0,2}$ to compute $F(X_{0,2})$ for all values of $F(X_{1,0})$ in ISF-1. Then, for $\Delta F(X_{2,0})$ associated to a pair $(x_1, x_2) \in$ ISF-1 and for $\Delta X_{3,0}$ associated to a pair $(y_1, y_2) \in$ ISF-2, combine the pairs to $\{(x_1, y_1), (x_2, y_2)\}$ if $\Delta F(X_{2,0}) = \Delta X_{3,0}$. For all the pairs in ISF-1 and all the pairs in ISF-2, store the combined pairs in a table named ISF-(1, 2). On average, ISF-(1,2) contains $2^{2b} \times 2^{2b} \times 2^{-b} = 2^{3b}$ pairs.

(3) Choose a random value for $X_{11,3}$ to compute $F(X_{8,6})$ for all values of $F(X_{9,4})$ in ISF-6. Then, for $\Delta F(X_{8,6})$ associated to a pair $(z_1, z_2) \in$ ISF-6 and for $\Delta X_{7,0}$ associated to a pair $(v_1, v_2) \in$ ISF-5, combine the pairs to $\{(z_1, v_1), (z_2, v_2)\}$ if $\Delta F(X_{8,6}) = \Delta X_{7,0}$. For all the pairs in ISF-6 and all the pairs in ISF-5, store the combined pairs in a table named ISF-(5, 6). On average, ISF-(5,6) contains $2^{2b} \times 2^{2b} \times 2^{-b} = 2^{3b}$ pairs.

(4) Choose a random value for $X_{8,5}$ to compute $F(X_{7,6})$ and $F(X_{6,0})$ for all the pairs in ISF-(5, 6). Then, discard the pairs from ISF-(5, 6) if $\Delta F(X_{6,0}) \neq \Delta X_{7,0}$. On average, ISF-(5, 6) contains $2^{3b} \times 2^{-b} = 2^{2b}$ pairs after this filtering.

(5) For $\Delta F(X_{5,0})$ associated to a pair $(u_1, u_2) \in$ ISF-3 and for $\Delta X_{6,0}$ associated to a pair $\{(z_1, v_1), (z_2, v_2)\} \in$ ISF-(5,6), combine the pairs to $\{(z_1, v_1, u_1), (z_2, v_2, u_2)\}$ if $\Delta F(X_{5,0}) = \Delta X_{6,0}$. For all the pairs in ISF-3 and all the pairs in ISF-(5, 6), store the combined pairs in a table named ISF-(3, 5, 6). On average, ISF-(3, 5, 6) contains $2^{2b} \times 2^{2b} \times 2^{-b} = 2^{3b}$ pairs.

(6) For all the pairs in ISF-(3, 5, 6), compute $X_{6,6} = F^{-1}(X_{7,6} + X_{5,0})$ and $X_{8,4} = F^{-1}(X_{7,6} + X_{9,4})$. Discard the pairs from ISF-(3, 5, 6) if $\Delta X_{6,6} \neq \Delta X_{8,4}$. On average, ISF-(3, 5, 6) contains $2^{3b} \times 2^{-b} = 2^{2b}$ pairs.

(7) Choose a random value for $X_{0,4}$ to compute $F(X_{1,2})$, $X_{2,2}$, and $X_{4,0}$ for the pairs in ISF-(1, 2). For $\Delta X_{4,0}$ associated to a pair $\{(x_1, y_1), (x_2, y_2)\} \in$ ISF-(1, 2) and for $\Delta X_{6,6}$ associated to a pair $\{(z_1, v_1, u_1), (z_2, v_2, u_2)\} \in$ ISF-(3, 5, 6), combine the pairs to $\{(x_1, y_1, z_1, v_1, u_1), (x_2, y_2, z_2, v_2, u_2)\}$ if $\Delta X_{4,0} = \Delta X_{6,6}$. For all the pairs in ISF-(1, 2) and all the pairs in ISF-(3, 5, 6), store the combined pairs in a table named ISF-(1, 2, 3, 5, 6). On average, ISF-(1, 2, 3, 5, 6) contains $2^{3b} \times 2^{2b} \times 2^{-b} = 2^{4b}$ pairs.

(8) For all the pairs in ISF-(1, 2, 3, 5, 6), compute $F(X_{4,0})$. Discard the pairs from ISF-(1, 2, 3, 5, 6) if $\Delta F(X_{4,0}) \neq \Delta X_{5,0}$. On average, ISF-(1, 2, 3, 5, 6) contains $2^{4b} \times 2^{-b} = 2^{3b}$ pairs after this filtering.

(9) For all the pairs in ISF-(1, 2, 3, 5, 6), compute $X_{5,6} = F^{-1}(X_{6,6} + X_{4,0})$ and $X_{4,6} = F^{-1}(X_{5,6} + X_{3,0})$. Discard the pairs from ISF-(1, 2, 3, 5, 6) if $\Delta X_{4,6} \neq \Delta X_{2,0}$. On average, ISF-(1, 2, 3, 5, 6) contains $2^{3b} \times 2^{-b} = 2^{2b}$.

(10) For $\Delta F(X_{5,4})$ associated to a pair $(q_1, q_2) \in$ ISF-4 and for $\Delta X_{4,6}$ associated to a pair $\{(x_1, y_1, z_1, v_1, u_1), (x_2, y_2, z_2, v_2, u_2)\} \in$ ISF-(1, 2, 3, 5, 6), combine the pairs to $\{(x_1, y_1, z_1, v_1, u_1, q_1), (x_2, y_2, z_2, v_2, u_2, q_2)\}$ if $\Delta F(X_{5,4}) = \Delta X_{4,6}$. For all the pairs in ISF-4 and all the pairs in ISF-(1, 2, 3, 5, 6), store the combined pairs in ISG2. On average, ISG2 contains $2^{2b} \times 2^{2b} \times 2^{-b} = 2^{3b}$.

(11) Compute the remaining parts. There are four filtering points with the ratio 2^{-b}. Therefore, after all computations, on average, ISG2 contains $2^{-4b} \times 2^{3b} = 2^{-b}$ pairs.

If the above procedure is repeated 2^b times, we expect to find one pair for the 11-round ISG2 of 8-branch GFN-2. And, the complexity of Step (8) is $2^{4b+1}F$ and much more dominant than the other steps. Therefore, the complexity of obtaining one pair for the ISG2 is $2^{5b+1}F\$$. The attacks based on the ISG2 are summarized in Table 6. The third attack in Table 6 is valid for $(a, b) = (4, 4), (8, 4)$, and $(8, 8)$. So, we denote the corresponding entry of the "(a, b)" column by $\neg(4, 8)$.

7. Conclusion

In this paper, we analyzed the security of GFN-2 in the known-key setting. We improved the results of 4-branch GFN-2 presented in \cite{KangHoMoKwSuHo12}. We also

presented the first known-key distinguishing and partial-collision attacks on 6-branch and 8-branch GFN-2 structures. We explained each attack such that the complexity and validity are easily understood. Our attacks do not mean that any block cipher with GFN-2 structure is insecure but can be useful and helpful in having an insight about the security of GFN-2 in known-key settings and in designing a new block cipher or hash function.

Data Availability

No data were used to support this study.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korea Government(MSIT) (no. B0722-16-0006, cross-layer design of cryptography and physical layer security for IoT networks).

References

[1] L. R. Knudsen and V. Rijmen, "Known-key distinguishers for some block ciphers," *ASIACRYPT 2007, LNCS 4833*, Springer, Berlin, Germany, pp. 315–324, 2007.

[2] F. Mendel, C. Rechberger, M. Schläffer, and S. S. Thomsen, "The rebound attack: cryptanalysis of reduced whirlpool and grøstl," *Fast Software Encryption, LNCS 5665*, Springer, Berlin, Germany, pp. 260–276, 2009.

[3] Y. Sasaki and K. Yasuda, "Known-key distinguishers on 11-round Feistel and collision attacks on its hashing modes," *Fast Software Encryption, LNCS 6733*, Springer, Berlin, Germany, pp. 397–415, 2011.

[4] H. Kang, D. Hong, D. Moon, D. Kwon, J. Sung, and S. Hong, "Known-key attacks on generalized Feistel schemes with SP round function," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E95.A, no. 9, pp. 1550–1560, 2012.

[5] H. Kang, D. Hong, J. Sung, and S. Hong, "Known-key attack on SM4 block cipher," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E100.A, no. 12, pp. 2985–2990, 2017.

[6] Y. Sasaki, S. Emami, D. Hong, and A. Kumar, "Improved known-key distinguishers on feistel-SP ciphers and application to Camellia," *Information Security and Privacy, LNCS 7372*, Springer, Berlin, Germany, pp. 87–100, 2012.

[7] FIPS, "Data encryption standard (DES)," FIPS 46-3, Oct. 25, 1999.

[8] ISO, *Information Technology—Security Techniques—Encryption Algorithms—Part 3: Block Ciphers*, ISO, Geneva, Switzerland, ISO/IEC 18033-3:2010, 2010.

[9] K. Aoki, T. Ichikawa, M. Kanda et al., "*Camellia*: a 128-bit block cipher suitable for multiple platforms—design and analysis," *SAC 2000, LNCS 2012*, Springer, Berlin, Germany, pp. 39–56, 2000.

[10] K. Nyberg, "Generalized Feistel networks," *ASIACRYPT'96, LNCS 1163*, Springer, Berlin, Germany, pp. 91–104, 1996.

[11] T. Shirai, K. Shibutani, T. Akishita, S. Moriai, and T. Iwata, "The 128-bit blockcipher CLEFIA (extended abstract)," *Fast Software Encryption, FSE 2007, LNCS*, Springer, Berlin, Germany, pp. 181–195, 2007.

[12] D. Hong, J. Sung, S. Hong et al., "HIGHT: a new block cipher suitable for low-resource device," *CHES 2006, LNCS 4249*, Springer, Berlin, Germany, pp. 46–59, 2006.

[13] B. Preneel, R. Govaerts, and J. Vandewalle, "Hash functions based on block ciphers: a synthetic approach," in *Proceedings of the CRYPTO'93*, pp. 368–378, Springer, Santa Barbara, CA, USA, August 1994.