Related-Key Attack on the Full HIGHT

Bonwook Koo, Deukjo Hong, and Daesung Kwon

The Attached Institute of ETRI
P.O. Box 1, Yuseong-Gu, Daejeon, Korea
{bwkoo,hongdj,ds_kwon}@ensec.re.kr

Abstract. HIGHT is a lightweight block cipher, proposed in CHES 2006 , and on the process of ISO/IEC 18033-3 standardization. It is a 32-round Feistel-like block cipher with 64-bit block and 128-bit key. In this paper, we present the first attack on the full HIGHT using related-key rectangle attack with $2^{123.169}$ encryptions, $2^{57.84}$ data, and 4 related keys. Our related-key rectangle attack is valid for 2^{126} weak keys and this attack can be easily extended to an attack for the full key space faster than an exhaustive key searching using 4 related keys.

We observe that an "add-difference" of master keys is propagated to an add-difference of subkeys with probability 1, so we can find 3-round local collisions of HIGHT by considering an add-difference as a relation of keys. Exploiting these local collisions and "over-simplified" structure of key-schedule, we construct a new 15.5-round related-key differential trail with relatively high probability. We construct a 24-round related-key rectangle distinguisher with probability $2^{-117.68}$ from an 8.5-round and a 15.5-round related-key truncated differential trail with local collisions by applying the ladder switch technique, and then suggest an attack on full rounds of HIGHT with this distinguisher. Our result implies that HIGHT cannot be regarded as an instantiation of the ideal cipher used in some provably secure schemes.

Keywords: HIGHT, Block cipher, Cryptanalysis, Related-key rectangle attack.

1 Introduction

In designing a block cipher, a strong key schedule has not been a main consideration. However, for recent years, the related-key attacks exploiting a weakness of a key schedule have provided interesting results [3, 4, 5, 7]. Most of them indicate that simple structure of a key schedule causes weakness useful for certain attacks. We have KASUMI and AES as such examples. KASUMI, known as the A5/3 algorithm for GSM security, has a linear key schedule. It is fully broken by the related-key rectangle attack [2] and practically broken by the related-key sandwich attack [7]. The key schedules of AES have a lot of symmetry in their structures and use at most four S-boxes to generate a 128-bit subkey for a round. So the full rounds of AES-192 and AES-256 are attacked by the related-key amplified boomerang and the related-key boomerang attacks, respectively [4]. A

K.-H. Rhee and D. Nyang (Eds.): ICISC 2010, LNCS 6829, pp. 49–67, 2011.

practical key recovery attack on 13 out of 14 rounds of AES-256, which has been recently proposed, also uses related keys [5].

HIGHT [8] is a block cipher which has a linear(in a modular addition point of view) key schedule with few propagations. It was proposed at CHES 2006 for lightweight computing environments such as radio frequency identifications (RFID). Also, HIGHT is a block cipher standard approved by Telecommunications Technology Association (TTA) of Korea and international standardization activities are in progress to include the HIGHT into ISO/IEC 18033-3 [9]. HIGHT is a 32-round block cipher in 8-branch type II generalized Feistel structure with 64-bit block and 128-bit key. The round functions of HIGHT is designed with bit-wise exclusive OR, addition modulo 2^8, and rotations. Such design aspects make HIGHT more efficient than most existing block ciphers including AES-128 on hardware implementation. The designers of HIGHT analyzed its security against various attacks including related-key attacks and they concluded that at least 20 rounds of HIGHT is secure against these attacks. But at ICISC 2007, Lu et al. presented some cryptanalytic results on the HIGHT reduced to 25, 26, and 28 rounds with or without initial and final whitening key additions, using impossible differential, related-key rectangle, related-key impossible differential attacks [10]. Moreover, at ACISP 2009, Özen et al. improved the attack results of ICISC 2007 into an impossible differential attack on 26 rounds of HIGHT and a related-key impossible differential attack on 31 rounds of HIGHT [13]. At CANS 2009, Zhang et al. pointed out an error in the 12-round saturation distinguisher introduced by designers of HIGHT and gave a saturation attack on 22 rounds of HIGHT with initial and final whitening keys using 17-round saturation distinguisher [15].

In this paper, we present a related-key attack on the full HIGHT slightly faster than the exhaustive key search. The attack consists of a related-key rectangle attack for a quarter of key space and an exhaustive key searching for the rest three-quarter of key space in the related-key attack model. Our related-key rectangle attack uses a 24-round related-key rectangle distinguisher with probability $2^{-117.68}$. This distinguisher is constructed from an 8.5-round($E0$) and a 15.5-round($E1$) related-key truncated differential trail by combining them with the ladder switch technique and $E1$ is a combination of three local collisions. The local collision is a related-key differential trail whose input and output differences are zero and in our attack, and we find two types of 4-round local collision and combine them alternately by using the byte-wise rotational property of subkey positioning. Every subkey byte is defined by a modular addition of a byte of encryption key and a predefined constant, so we give an *add-difference* for relation of keys to avoid paying probability for generating subkey differences by key schedule. For $E0$, we modify a known related-key differential trail [10,11] to avoid a flaw shown in Appendix A and transform it into related-key truncated differential trail to reduce data complexity. So we construct a related-key rectangle attack for a quarter of key space with $2^{123.17}$ time and $2^{57.84}$ data and an attack for whole key space with $2^{125.833}$ time and $2^{57.84}$ data. The time and data complexities for attacking HIGHT is given in the Table 1.

Table 1. Summary of the attacks on HIGHT(Imp.:Impossible, Diff.:Differential, Rel.:Related, Rec.:Rectangle, Wek.:Weak Key)

Rounds	Attack	Complexities		References
		Data	Time	
18	Imp. Diff.	$2^{46.8}$	$2^{109.2}$	[8]
22	Saturation	$2^{62.04}$	$2^{118.71}$	[15]
25	Imp. Diff.	2^{60}	$2^{126.78}$	[10]
26	Imp. Diff.	2^{61}	$2^{119.53}$	[13]
26	Rel.-Key Rec.	$2^{51.2}$	$2^{120.41}$	[10]
28	Rel.-Key Imp.	2^{60}	$2^{125.54}$	[10]
31	Rel.-Key Imp.	2^{63}	$2^{127.28}$	[13]
32	Rel.-Key Rec. for Wek.	$2^{57.84}$	$2^{123.17}$	This paper
32	Rel.-Key attack	$2^{57.84}$	$2^{125.833}$	This paper

To our knowledge, this is the first cryptanalytic result on the full rounds of HIGHT. Our attack has very high complexity but clearly, it shows that HIGHT does not reach the security goal in the related-key attack model, required for a block cipher which has a 64-bit block and a 128-bit key. It is also an evidence that HIGHT cannot be regarded as an instantiation of an ideal cipher. Namely, HIGHT would not be used as a substitute for an ideal cipher in applications which are provably secure based on ideal cipher, e.g., some block-cipher-based hash function schemes.

This paper is organized as follows. Section 2 gives the specifications of HIGHT. In Section 3, two types of local collisions of HIGHT and its probability are introduced and a weak key space is classified. In Section 4, a 24-round related-key rectangle distinguisher of HIGHT is presented with its separation into $E0$ and $E1$ and estimation of its probability. Attack procedure and complexity analysis are shown in Section 5. Finally, Section 6 concludes this paper. In Appendix A, some flaws in calculating a probability of differential trail include key addition is pointed out. A complexity of exhaustive key searching in related-key model is presented in Appendix B. An overall view of our attack is depicted in Appendix C.

2 Description of HIGHT

Throughout this paper, we use the following notations.

- $\oplus$: bitwise exclusive OR(XOR)
- $\boxplus$: addition modulo 2^8
- $\Delta(\nabla)$: a notation of xor-difference, an xor-difference Δx indicates that a pair $(x, x')$is defined by $x' = x \oplus \Delta x$
- $\Delta^+(\nabla^+)$: a notation of add-difference, an add-difference $\Delta^+ x$ indicates that a pair $(x, x')$is defined by $x' = x \boxplus \Delta^+ x$

- $X[i_1, i_2, ..., i_n]$: concatenation of $X[i_1]$, $X[i_2]$, ..., and $X[i_n]$
- $MSB_i(X)$: the most significant i bits of a string X
- $LSB_i(X)$: the least significant i bits of a string X
- $(\Delta X, \Delta Y) \xrightarrow{\boxplus} \Delta Z$: an event that $(x \boxplus y) \oplus (x' \boxplus y') = \Delta Z$, where $x \oplus x' = \Delta X$ and $y \oplus y' = \Delta Y$

HIGHT takes a 64-bit plaintext P and a 128-bit key K, and its 32-round encryption procedure produces a 64-bit ciphertext C. From now on, we present any 64-bit variable A and any 128-bit variable B as a tuple of eight bytes $(A[7], ..., A[1], A[0])$ and a tuple of sixteen bytes $(B[15], ..., B[1], B[0])$.

The key schedule produces 128 8-bit subkeys $\mathrm{SK}[0], ..., \mathrm{SK}[127]$ from a 128-bit key $K = (K[15], ..., K[0])$: for $0 \le i \le 7$ and $0 \le j \le 7$,

$$\begin{cases} \mathrm{SK}[16i + j] \leftarrow K[j - i \bmod 8] \boxplus \delta[16i + j], \\ \mathrm{SK}[16i + j + 8] \leftarrow K[(j - i \bmod 8) + 8] \boxplus \delta[16i + j + 8], \end{cases}$$

where $\delta[0], ..., \delta[127]$ are public constants.

Let $X_{i-1} = (X_{i-1}[7], ..., X_{i-1}[0])$ and $X_i = (X_i[7], ..., X_i[0])$ be the input and output of the round $i - 1$ for $1 \le i \le 32$, respectively, where 'round i' denotes the $(i + 1)$-th round(i.e. round 0 implies the first round).

The encryption procedure of HIGHT is as follows.

1. Initial Transformation:

$$\begin{aligned} &X_0[0] \leftarrow P[0] \boxplus K[12]; X_0[2] \leftarrow P[2] \oplus K[13]; \\ &X_0[4] \leftarrow P[4] \boxplus K[14]; X_0[6] \leftarrow P[6] \oplus K[15]; \\ &X_0[1] \leftarrow P[1]; X_0[3] \leftarrow P[3]; X_0[5] \leftarrow P[5]; X_0[7] \leftarrow P[7]. \end{aligned}$$

2. Round Iteration for $1 \le i \le 32$:

$$\begin{aligned} &X_i[0] \leftarrow X_{i-1}[7] \oplus (F_0(X_{i-1}[6]) \boxplus \mathrm{SK}[4i - 1]); \\ &X_i[2] \leftarrow X_{i-1}[1] \boxplus (F_1(X_{i-1}[0]) \oplus \mathrm{SK}[4i - 2]); \\ &X_i[4] \leftarrow X_{i-1}[3] \oplus (F_0(X_{i-1}[2]) \boxplus \mathrm{SK}[4i - 3]); \\ &X_i[6] \leftarrow X_{i-1}[5] \boxplus (F_1(X_{i-1}[4]) \oplus \mathrm{SK}[4i - 4]); \\ &X_i[1] \leftarrow X_{i-1}[0]; X_i[3] \leftarrow X_{i-1}[2]; X_i[5] \leftarrow X_{i-1}[4]; X_i[7] \leftarrow X_{i-1}[6], \end{aligned}$$

   where bijective linear functions F_0 and F_1 are defined by

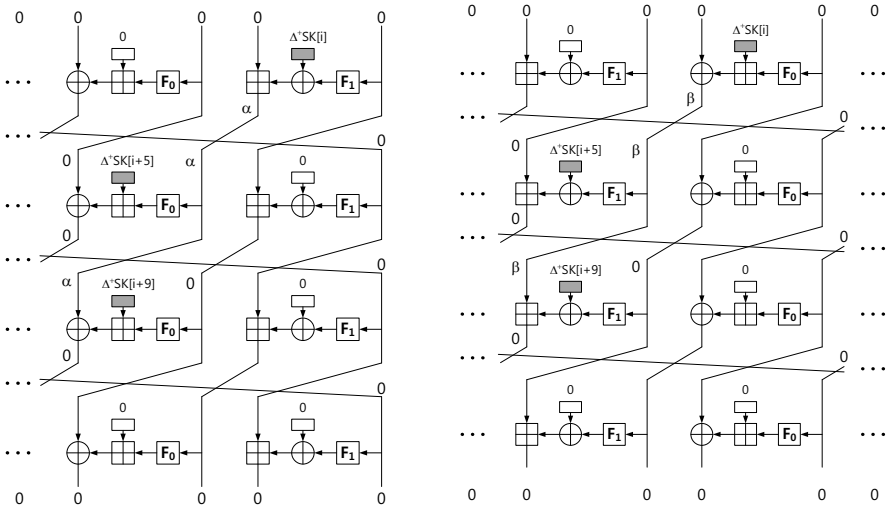
$$\begin{cases} F_0(x) = x^{\lll 1} \oplus x^{\lll 2} \oplus x^{\lll 7}, \\ F_1(x) = x^{\lll 3} \oplus x^{\lll 4} \oplus x^{\lll 6}. \end{cases}$$

3. Final Transformation:

$$\begin{aligned} &C[0] \leftarrow X_{32}[1] \boxplus K[0]; C[2] \leftarrow X_{32}[3] \oplus K[1]; \\ &C[4] \leftarrow X_{32}[5] \boxplus K[2]; C[6] \leftarrow X_{32}[7] \oplus K[3]; \\ &C[1] \leftarrow X_{32}[2]; C[3] \leftarrow X_{32}[4]; C[5] \leftarrow X_{32}[6]; C[7] \leftarrow X_{32}[0]. \end{aligned}$$

3 Local Collisions in HIGHT

Local collision is firstly introduced by Chabaud et al. in [6] for finding collisions in SHA-0 hash function using differential cryptanalysis. In block cipher cryptanalysis, if a difference caused only by a subkey difference is eliminated by other subkey differences with some probability a few rounds later, we call this property a local collision in block cipher. In HIGHT, we observe that there are two types of local collision which are depicted in Fig. 1.



(a) Local collision type A **(b)** Local collision type B

Fig. 1. Local collisions in HIGHT

3.1 Probabilities of Local Collisions

Fig. 1-(a) shows how the only nonzero differences $\Delta^+SK[i]$, $\Delta^+SK[i+5]$, and $\Delta^+SK[i+9]$ in the form of the local collision type A lead to zero output differences of the round. Its probability is computed as $r = \sum_\alpha r_1(\alpha)r_2(\alpha)r_3(\alpha)$ where

$$
\begin{aligned}
r_1(\alpha) &= \Pr(((X \oplus Y) + Z) \oplus ((X \oplus (Y + \Delta^+SK[i])) + Z) = \alpha),\\
r_2(\alpha) &= \Pr((X + Y) \oplus ((X \oplus F_0(\alpha)) + (Y + \Delta^+SK[i+5])) = 0),\\
r_3(\alpha) &= \Pr((X + Y) \oplus (X + (Y + \Delta^+SK[i+9])) = \alpha).
\end{aligned}
$$

Similarly, Fig. 1-(b) shows how the only nonzero differences $\Delta^+SK[i]$, $\Delta^+SK[i+5]$, and $\Delta^+SK[i+9]$ in the form of the local collision type B lead to zero output differences of the round. Its probability is computed as $s = \sum_\beta s_1(\beta)s_2(\beta)s_3(\beta)$ where

$$s_1(\beta) = \Pr((X+Y) \oplus (X + (Y + \Delta^+SK[i])) = \beta),$$
$$s_2(\beta) = \Pr(((X \oplus Y) + Z) \oplus (((X \oplus F_1(\beta)) \oplus (Y + \Delta^+SK[i+5])) + Z) = 0),$$
$$s_3(\beta) = \Pr(((X \oplus Y) + Z) \oplus ((X \oplus (Y + \Delta^+SK[i+9])) + (Z \oplus \beta)) = 0).$$

By an exhaustive computation, we can see the expected values of r and s when $(\Delta^+SK[i], \Delta^+SK[i+5], \Delta^+SK[i+9]) = (\texttt{0x10}, \texttt{0x68}, \texttt{0x10})$ are $2^{-5.1420}$ and 2^{-8}, respectively, and we the following 8 possibilities of $(\Delta^+SK[i], \Delta^+SK[i+5], \Delta^+SK[i+9])$ yielding the same probabilities:

$$(\texttt{0x10}, \texttt{0x68}, \texttt{0x10}), (\texttt{0x10}, \texttt{0x68}, \texttt{0xf0}), (\texttt{0xf0}, \texttt{0x68}, \texttt{0x10}), (\texttt{0xf0}, \texttt{0x68}, \texttt{0xf0}),$$
$$(\texttt{0x10}, \texttt{0x98}, \texttt{0x10}), (\texttt{0x10}, \texttt{0x98}, \texttt{0xf0}), (\texttt{0xf0}, \texttt{0x98}, \texttt{0x10}), (\texttt{0xf0}, \texttt{0x98}, \texttt{0xf0}).$$

We observed that the probability r that the local collision type A with $(\Delta^+SK[i], \Delta^+SK[i+5], \Delta^+SK[i+9]) = (\texttt{0x10}, \texttt{0x68}, \texttt{0x10})$ occurs is nonzero only when $\alpha = \texttt{0x10}$ or $\texttt{0x30}$. Under the observation, the probability r is actually $2^{-4.67807}$, $2^{-5.41504}$, or $2^{-6.41504}$. So, we regard $2^{-6.41504}$ as a lower bound of r.

Similarly, we observed that the probability s that the local collision type B with $(\Delta^+SK[i], \Delta^+SK[i+5], \Delta^+SK[i+9]) = (\texttt{0x10}, \texttt{0x68}, \texttt{0x10})$ occurs is nonzero only when $\beta = \texttt{0x70}$. Especially, for $\beta = \texttt{0x70}$, $s_2(\beta)$ is nonzero only when

$$SK[i+5] \in T = \{x \vee \texttt{0x18} | x \in \mathrm{GF}(2^8)\}.$$

When $SK[i+5] \in T$, the local collision of type B occurs with the probability $s = 2^{-6}$. Otherwise, it does not occur. Note that the fraction of T in $\mathrm{GF}(2^8)$ is $1/4$.

3.2 Local Collisions to a Long Differential Trail

We can use a sequence of local collisions, 'type A – type B – type A' to construct a 12-round related-key differential trail. Let i be a multiple of 4(i.e. $\Delta^+SK[i]$ be the right most subkey difference of round $i/4$). If $\Delta^+SK[i]$, $\Delta^+SK[i+5]$, and $\Delta^+SK[i+9]$ are induced by the only nonzero add-differences $\Delta^+K[j_1]$, $\Delta^+K[j_2]$, and $\Delta^+K[j_3]$ of master-key bytes, then by rotational property of key schedule,

$$\Delta^+K[j_1] = \Delta^+SK[i] = \Delta^+SK[i+17] = \Delta^+SK[i+34],$$
$$\Delta^+K[j_2] = \Delta^+SK[i+5] = \Delta^+SK[i+22] = \Delta^+SK[i+39],$$
$$\Delta^+K[j_3] = \Delta^+SK[i+9] = \Delta^+SK[i+26] = \Delta^+SK[i+43],$$

and differences of other subkeys are all zero if differences of other master key bytes are zero.

Therefore, if there exist nonzero add-differences $\Delta^+K[j_1]$, $\Delta^+K[j_2]$, and $\Delta^+K[j_3]$ such that the probabilities p_1, p_2, and p_3 of local collisions from $i/4$ to $(i/4+3)$-th round, from $(i/4+4)$ to $(i/4+7)$-th round, and from $(i/4+8)$ to $(i/4+11)$-th round are all nonzero, then we can find a 12-round related-key differential trail of HIGHT with probability $p_1 \times p_2 \times p_3$ by combining them sequentially.

From the arguments in Section 3.1, when we take $(\Delta^+K[j_1], \Delta^+K[j_2], \Delta^+K[j_3])$ = (0x10, 0x68, 0x10), the 12-round related-key differential trail is valid only for a quarter of the whole key space and its probability is lower bounded by $2^{-18.83008}$.

4 Related-Key Rectangle Distinguisher for 24 Rounds of HIGHT

4.1 Related-Key Rectangle Distinguisher

A rectangle distinguisher assumes that a block cipher $E_K : \{0,1\}^n \to \{0,1\}^n$ with an arbitrary key K can be represented by a composition of two sub-ciphers $E0_K$ and $E1_K$, i.e. $E_K = E1_K \circ E0_K$, where n is the bit-length of block. Our approach to construct a related-key rectangle distinguisher is somewhat different from previous works in the point that we use xor-difference for plaintexts or ciphertexts and add-difference for keys.

Assume that we have two related-key differentials for $E0$ and $E1$ with the following probabilities

$$p = \Pr[E0_K(P) \oplus E0_{K \boxplus \Delta^+ K}(P \oplus \Delta P) = \Delta Y], \tag{1}$$
$$q = \Pr[E1_K(Y) \oplus E1_{K \boxplus \nabla^+ K}(Y \oplus \nabla Y) = \nabla C]. \tag{2}$$

We consider four encryption oracles with 4 related keys denoted by E_{K1}, E_{K2}, E_{K3}, and E_{K4} and the relations between keys are as follows,

$$K2 = K1 \boxplus \Delta^+ K, \qquad K4 = K3 \boxplus \Delta^+ K,$$
$$K3 = K1 \boxplus \nabla^+ K, \qquad K4 = K2 \boxplus \nabla^+ K.$$

For a plaintext quartet (P_1, P_2, P_3, P_4) such that $P_1 \oplus P_2 = P_3 \oplus P_4 = \Delta P$, let $Y_i = E0_{Ki}(P_i)$ and $C_i = E_{Ki}(P_i) = E1_{Ki}(Y_i)$ for $1 \le i \le 4$. If the event $Y_1 \oplus Y_2 = Y_3 \oplus Y_4 = \Delta Y$ and the event $Y_1 \oplus Y_3 = \nabla Y$ occur, we obtain $Y_2 \oplus Y_4 = \nabla Y$ because

$$\begin{aligned} Y_2 \oplus Y_4 &= (Y_2 \oplus Y_1) \oplus (Y_1 \oplus Y_3) \oplus (Y_3 \oplus Y_4) \\ &= \Delta Y \oplus \nabla Y \oplus \Delta Y = \nabla Y. \end{aligned}$$

Therefore, for a randomly chosen plaintext quartet (P_1, P_2, P_3, P_4) such that $P_1 \oplus P_2 = P_3 \oplus P_4 = \Delta P$, we have $C_1 \oplus C_3 = C_2 \oplus C_4 = \nabla C$ with the probability $p^2 \cdot 2^{-n} \cdot q^2$, from (1) and (2). If there exist more than two values for ΔY and ∇Y, the probability is amplified to

$$\hat{p}^2 \cdot 2^{-n} \cdot \hat{q}^2, \quad \text{where} \quad \hat{p}^2 = \sum_{\Delta Y} p^2 \quad \text{and} \quad \hat{q}^2 = \sum_{\nabla Y} q^2. \tag{3}$$

Our attack assumes more than two values for ΔP so our probability calculation in the next section would be slightly differ from (3).

4.2 Related-Key Rectangle Distinguisher of HIGHT

Related-key differential trail for $E0$ is based on a trail introduced in [10,11] but significantly modified to avoid the flaw explained in Appendix A and changed into truncated differential to reduce the data complexity. Related-key differential trail for $E1$ includes three local collisions as described in Section 3.1.

We define $E0$ and $E1$ by partial rounds from round 3 to round 10.5 and round 10.5 to round 26, respectively(0.5 round implies computation of 2 round functions out of 4 round functions in a round). The input and output bytes to $E0$ and $E1$ and corresponding differences are described in Table 2, where the $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ are defined by sets of hexadecimal values as follows,

$$\mathcal{A} = \{\texttt{14}, \texttt{1c}, \texttt{24}, \texttt{2c}, \texttt{34}, \texttt{3c}, \texttt{54}, \texttt{5c}, \texttt{64}, \texttt{6c}, \texttt{74}, \texttt{7c}, \texttt{d4}, \texttt{dc}, \texttt{e4}, \texttt{ec}\},$$
$$\mathcal{B} = \{\texttt{14}, \texttt{1c}, \texttt{24}, \texttt{2c}, \texttt{34}, \texttt{3c}, \texttt{54}, \texttt{5c}, \texttt{64}, \texttt{6c}, \texttt{74}, \texttt{7c}, \texttt{d4}, \texttt{dc}, \texttt{e4}, \texttt{ec}, \texttt{f4}, \texttt{fc}\},$$
$$\mathcal{C} = \{\texttt{10}, \texttt{30}, \texttt{70}, \texttt{f0}\}.$$

Table 2. Byte positions and differences of both inputs and outputs for distinguishers of $E0$ and $E1$

$E0$	Positions	Input	$(X_3[7], X_3[6], X_3[5], X_3[4], X_3[3], X_3[2], X_3[1], X_3[0])$
		Output	$(X_{12}[7], X_{11}[5], X_{10}[3], X_{10}[2], X_{11}[2], X_{12}[2], X_{13}[2], X_{13}[1])$
	Differences	Input	(0x0, 0x0, $\mathcal{A}$, 0x80, 0x0, 0x0, 0x0, 0x0)
		Output	(0x0, 0x0, 0x0, $\mathcal{B}$, 0x80, 0x0, 0x0, 0x0)
Δ^+K	(0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x80, 0x0)		
$E1$	Positions	Input	$(X_{12}[7], X_{11}[5], X_{10}[3], X_{10}[2], X_{11}[2], X_{12}[2], X_{13}[2], X_{13}[1])$
		Output	$(X_{27}[7], X_{27}[6], X_{27}[5], X_{27}[4], X_{27}[3], X_{27}[2], X_{27}[1], X_{27}[0])$
	Differences	Input	(0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, $\mathcal{C}$)
		Output	(0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x0)
∇^+K	(0x0, 0x0, 0x0, 0x0, 0x0, 0x0, 0x10, 0x0, 0x0, 0x0, 0x68, 0x0, 0x0, 0x0, 0x10, 0x0)		

The related-key differential trails for $E0$ and $E1$ are depicted in the the following fig. 5 in Appendix C.

Probability of Related-Key Differential Trail for $E0$. Our attack begins with gathering plaintext pairs which satisfy $\Delta X_3[5] \in \mathcal{A}$, $\Delta X_3[4] = \texttt{0x80}$, and $\Delta X_3[0,1,2,3,6,7] = \texttt{0x0}$. Let a_i denote each element in $\mathcal{A}$, where $i = 0, 1, ..., 15$. The number of pairs such that $\Delta X_3[5] = a_i$ is same to the number of pairs such that $\Delta X_3[5] = a_j$ for all $0 \leq i, j \leq 15$. Let u_i denote the probabilities that $\Delta X_3[5] = a_i$ and $\Delta X_4[6] = 0$ for $i = 0, 1, ..., 15$, then

$$u_i = \Pr[\Delta X_4[6] = 0 | \Delta X_3[5] = a_i] \times \Pr[\Delta X_3[5] = a_i],$$

and since $\Pr[\Delta X_3[5] = a_i] = 2^{-4}$ for all $i = 0, 1, ..., 15$, the probability $\bar{u}$ such that $\Delta X_4[6] = 0$, among prepared pairs is calculated by

$$\bar{u} = \sum_{i=0}^{15} u_i = \frac{1}{16} \sum_{i=0}^{15} \Pr[\Delta X_4[6] = 0 | \Delta X_3[5] = a_i] > 2^{-4.09312}.$$

Let b_i denote each element in $\mathcal{B}$ and v_i denote the probabilities that $\Delta X_{10}[2] = b_i$ for $i = 0, 1, ..., 17$, then the probability $\hat{p}^2$ that both related-key differential trails for $E0$ are satisfied with the same output difference is calculated by

$$\hat{p}^2 = \bar{u}^2 \cdot \sum_{i=0}^{17} v_i^2 > 2^{-8.18624} \times 2^{-3.83007} > 2^{-12.017}.$$

Probability of Related-Key Differential Trail for $E1$ For each element c_i in $\mathcal{C}$, let the probabilities w_i be defined by

$$w_i = \Pr[\Delta X_{14}[2] = 0 | \Delta X_{13}[1] = c_i],$$

for $i = 0, 1, 2, 3$, then $\Delta X_{14}[0, 1, ..., 8] = 0$ with probabilities w_i. Both w_3 and w_4 are 2^{-3} for all $SK[52]$, whereas w_1 and w_2 are among 2^{-1}, 2^{-2}, and 2^{-3} according to $SK[52]$. So the lower-bound of $w_i (i = (0, 1, 2, 3)$ is 2^{-3}.

Since we assume that $\nabla^+ K[1] = \texttt{0x10}$, $\nabla^+ K[5] = \texttt{0x68}$, and $\nabla^+ K[9] = \texttt{0x10}$, we can calculate a nonzero probability q such that three local collisions occur sequentially as described in Section 3.1. As we know that both the first and the third local collisions during round 15∼18 and round 23∼26 are of type A and their probabilities are bounded below by $2^{-6.41504}$ and the second local collision during round 19∼22 is of type B and its probability is 2^{-6}, the probability q such that related-key differential trail from round 15 to 26 is calculated by

$$2^{-6.41504-6-6.41504} = 2^{-18.83008} < q.$$

Hence, the probability $\hat{q}^2$ that both related-key differential trails for $E1$ are satisfied is calculated by

$$\hat{q}^2 = \sum_{i=0}^{3} w_i^2 \cdot q^2 \geq 2^{-4} \times 2^{-37.66016} = 2^{-41.66016} > 2^{-41.661}.$$

Therefore, we have a 24-round related-key rectangle distinguisher with the probability

$$\hat{p}^2 \cdot 2^{-64} \cdot \hat{q}^2 \geq 2^{-12.017-64-41.661} = 2^{-117.678} > 2^{-117.68}.$$

The probabilities occurring by additions between differences are computed by exhaustive counting with PC. By experiments on PC, we make sure that suggested probabilities of related-key differential trail for $E0$ and $E1$ are lower bounds of the actual ratio of right pairs for $E0$ and $E1$ respectively, under the assumption that plaintexts and related keys are randomly chosen.

5 Related-Key Rectangle Attack for the Full Rounds of HIGHT

In this section, we describe the attack for the full rounds of HIGHT by using the 24-round related-key rectangle distinguisher explained in Section 4.1 for round 3 to round 26. However, note that the distinguisher is valid only for a quarter of the key space. So, we apply a related-key rectangle attack for a quarter of the key space and an exhaustive key search for the other part of the key space. The outline of our attack is as follows.

1. **Related-key rectangle attack**: We denote the set of the key quartets by $\mathcal{K}_1$ such that the 24-round related-key rectangle distinguisher in Section 4.1 is valid. Assuming that we are given a key quartet from $\mathcal{K}_1$, we perform a related-key rectangle attack which consists of the following phases.
   (a) **Constructing the plaintext set**: We construct the plaintext set $\mathcal{S}$ for extracting the plaintext quartets required for the related-key rectangle distinguisher.
   (b) **Guessing and filtering**: Let Z_1 be required key bits to check whether a plaintext quartet from the plaintext set $\mathcal{S}$ satisfies the input differences of the distinguisher. We guess a value z_1 for Z_1 and select the plaintext quartets from $\mathcal{S}$ satisfying the input differences of the distinguisher with z_1. Then, we discard the quartets whose ciphertext differences do not match with the output differences of the distinguisher.
   (c) **Counting and sorting**: Let Z_2 be required key bits to check whether a surviving quartet satisfies the output differences of the distinguisher. For each candidate (z_1, z_2) for (Z_1, Z_2), we count the number of quartets satisfying the output differences of the distinguisher and restore it to the counter $t_{(z_1,z_2)}$. We sort the list of (z_1, z_2) according to $t_{(z_1,z_2)}$.
   (d) **Searching with the list**: We exhaustively search for the remained key bits for candidates with remarkably high $t_{(z_1,z_2)}$ until a right key quartet is found. If no right key quartet is found, go to (b) **Guessing and filtering phase**.
2. **Exhaustive key searching**: We denote the key space of HIGHT by $\mathcal{K}$ and let $\mathcal{K}_2 = \mathcal{K} \backslash \mathcal{K}_1$. Unless we find a right key quartet in $\mathcal{K}_1$ in the way of the related-key rectangle attack phase, we try to search it exhaustively for $\mathcal{K}_2$ in the way described in Appendix B.

5.1 Attack Procedure

Let sets $\mathcal{D}, \mathcal{E}, \mathcal{F}$, and $\mathcal{G}$ be defined by

$$\begin{aligned}
\mathcal{D} &= \{x \vee \texttt{0x18} | \forall x \in \mathrm{GF}(2^8)\}, \\
\mathcal{E} &= \{\texttt{0x00}, \texttt{0x20}, \texttt{0x40}, \texttt{0x60}, \texttt{0x80}, \texttt{0xa0}, \texttt{0xc0}, \texttt{0xe0}\}, \\
\mathcal{F} &= \{\texttt{0x18}, \texttt{0x28}, \texttt{0x38}, ..., \texttt{0xf8}\}, \\
\mathcal{G} &= \{\texttt{0x10}, \texttt{0x30}, \texttt{0x70}, \texttt{0xf0}\}.
\end{aligned}$$

Constructing the plaintext set

1. Choose $58657 \approx 2^{15.84}$ structures $\mathcal{S}_i$ of 2^{40} plaintexts ${}^i_lP$ each, $i = 1, 2, ..., 58657, l = 1, 2, ..., 2^{40}$, where in each structure, the 0, 6, 7-th bytes of ${}^i_lP$ are fixed, and the remaining 5 bytes take all the possible values. Obtain the ciphertexts ${}^i_lC$, ${}^i_lC^*$, ${}^i_lC'$, and ${}^i_lC'^*$ of ${}^i_lP$ encrypted with four related keys $K1$, $K2$, $K3$, and $K4$ respectively, where keys have a relation described in Table 2 of Section 4.1.

Guessing and filtering

2. Guess the 9 bytes $K1[0, 1, 2, 5, 6, 10, 12, 13, 14]$ such that $SK1[82] \in \mathcal{D}$ and do as follows, where SKi is subkey bytes produced by a secret key Ki.

   (a) Compute the subkeys $SK1[0, 1, 2, 5, 6, 10]$, and their related subkeys. Partially encrypt plaintext bytes ${}^i_lP[1, 2, 3, 4, 5]$ for each ${}^i_lP$ through partial rounds 0, 1, and 2 with 5 guessed subkey bytes and its related subkey bytes to get the following sets of intermediate values,

$$\{{}^i_lX_2[3], {}^i_lX_3[5], {}^i_lX_3[6], {}^i_lX_2[6], {}^i_lX_2[7]\},$$
$$\{{}^i_lX^*_2[3], {}^i_lX^*_3[5], {}^i_lX^*_3[6], {}^i_lX^*_2[6], {}^i_lX^*_2[7]\},$$
$$\{{}^i_lX'_2[3], {}^i_lX'_3[5], {}^i_lX'_3[6], {}^i_lX'_2[6], {}^i_lX'_2[7]\},$$
$$\{{}^i_lX'^*_2[3], {}^i_lX'^*_3[5], {}^i_lX'^*_3[6], {}^i_lX'^*_2[6], {}^i_lX'^*_2[7]\},$$

   for all $1 \le i \le 2^{15.84}$ and $1 \le l \le 2^{40}$.

   (b) Find all pairs $({}^i_lP, {}^i_uP)$ such that ${}^i_lX_2[3] \oplus {}^i_uX^*_2[3] = \texttt{0x80}$, ${}^i_lX_3[6] \oplus {}^i_uX^*_3[6] = {}^i_lX_2[6] \oplus {}^i_uX^*_2[6] = {}^i_lX_2[7] \oplus {}^i_uX^*_2[7] = \texttt{0x0}$, and ${}^i_lX_3[5] \oplus {}^i_uX^*_3[5] \in \mathcal{A}$ and store the corresponding ciphertext pairs $({}^i_lC, {}^i_uC^*)$ encrypted with each $K1$ and $K2$ in a hash table $\mathcal{H}$, for all $1 \le i \le 2^{15.84}$.

   (c) Find all pairs $({}^j_vP, {}^j_wP)$ such that ${}^j_vX'_2[3] \oplus {}^j_wX'^*_2[3] = \texttt{0x80}$, ${}^j_vX'_3[6] \oplus {}^j_wX'^*_3[6] = {}^j_vX'_2[6] \oplus {}^j_wX'^*_2[6] = {}^j_vX'_2[7] \oplus {}^j_wX'^*_2[7] = \texttt{0x0}$, and ${}^j_vX'_3[5] \oplus {}^j_wX'^*_3[5] \in \mathcal{A}$ and store the corresponding ciphertext pairs $({}^j_vC', {}^j_wC'^*)$ encrypted with each $K3$ and $K4$ in a hash table $\mathcal{I}$, for all $1 \le j \le 2^{15.84}$.

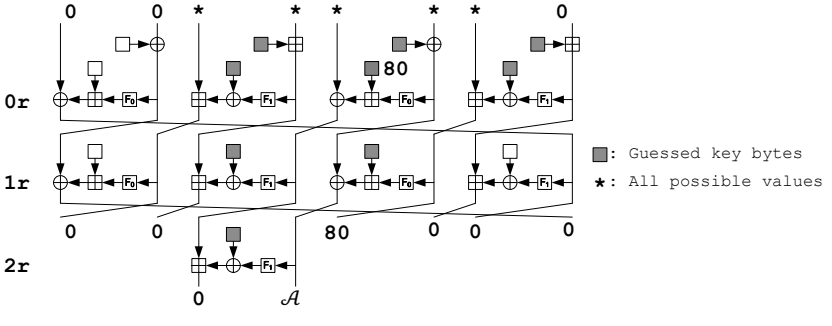


Fig. 2. Constructing plaintext sets and choosing pairs

(d) Store all quartets $({}^{i}_{l}C, {}^{i}_{u}C^{*}, {}^{j}_{v}C', {}^{j}_{w}C'^{*})$ defined by all pairs $({}^{i}_{l}C, {}^{i}_{u}C^{*}) \in \mathcal{H}$ and $({}^{j}_{v}C', {}^{j}_{w}C'^{*}) \in \mathcal{I}$, for all $1 \le i, j \le 2^{15.84}$ and $1 \le l, u, v, w \le 2^{40}$ in a hash table $\mathcal{J}$.

(e) For all quartets $({}^{i}_{l}C, {}^{i}_{u}C^{*}, {}^{j}_{v}C', {}^{j}_{w}C'^{*})$ in $\mathcal{J}$, do filtering by the following steps. In each steps, discard the quartets which do not satisfy the conditions and if less than 3 quartets are remained, then go to Step 2 with another key guessing.

i. Check if ${}^{i}_{l}C[0,1] \oplus {}^{j}_{v}C'[0,1] = 0$, ${}^{i}_{u}C^{*}[0,1] \oplus {}^{j}_{w}C'^{*}[0,1] = 0$, ${}^{i}_{l}C[2] \oplus {}^{j}_{u}C'[2] \in \mathcal{E}$, and ${}^{i}_{v}C^{*}[2] \oplus {}^{j}_{w}C'^{*}[2] \in \mathcal{E}(2^{-42}$ filtering).

ii. Compute and check if ${}^{i}_{l}X_{31}[3] \oplus {}^{j}_{u}X'_{31}[3] \in \mathcal{F}$, and ${}^{i}_{v}X^{*}_{31}[3] \oplus {}^{j}_{w}X'^{*}_{31}[3] \in \mathcal{F}(2^{-8}$ filtering).

iii. Check if $\Pr[(F_0({}^{i}_{l}C[6] \oplus {}^{j}_{u}C'[6]), 0) \xrightarrow{\boxplus} {}^{i}_{l}C[7] \oplus {}^{j}_{u}C'[7]] > 0$ and $\Pr[(F_0({}^{i}_{v}C^{*}[6] \oplus {}^{j}_{w}C'^{*}[6]), 0) \xrightarrow{\boxplus} {}^{i}_{v}C^{*}[7] \oplus {}^{j}_{w}C'^{*}[7]] > 0 (2^{-5.65514}$ filtering).

iv. Compute $\Delta T = {}^{i}_{l}X_{30}[4] \oplus {}^{j}_{u}X'_{30}[4]$ and $\Delta T' = {}^{i}_{v}X^{*}_{30}[4] \oplus {}^{j}_{w}X'^{*}_{30}[4]$, and check if $\Pr[(F_1(\Delta T), 0) \xrightarrow{\boxplus} {}^{i}_{l}C[6] \oplus {}^{j}_{u}C'[6]] > 0$ and $\Pr[(F_1(\Delta T'), 0) \xrightarrow{\boxplus} {}^{i}_{v}C^{*}[6] \oplus {}^{j}_{w}C'^{*}[6]] > 0$ $(2^{-5.65514}$ filtering).

v. Compute and check if $\Pr[(F_1({}^{i}_{l}X_{31}[4] \oplus {}^{j}_{u}X'_{31}[4]), \Delta T) \xrightarrow{\boxplus} {}^{i}_{l}C[5] \oplus {}^{j}_{u}C'[5]] > 0$ and $\Pr[(F_1({}^{i}_{v}X^{*}_{31}[4] \oplus {}^{j}_{w}X'^{*}_{31}[4]), \Delta T') \xrightarrow{\boxplus} {}^{i}_{v}C^{*}[5] \oplus {}^{j}_{w}C'^{*}[5]] > 0$ $(2^{-4.69704}$ filtering).

vi. If 3 or more quartets $({}^{i}_{l}C, {}^{i}_{u}C^{*}, {}^{j}_{v}C', {}^{j}_{w}C'^{*})$ remained, record them and go to Step 3; otherwise, go to Step 2 with another guess.

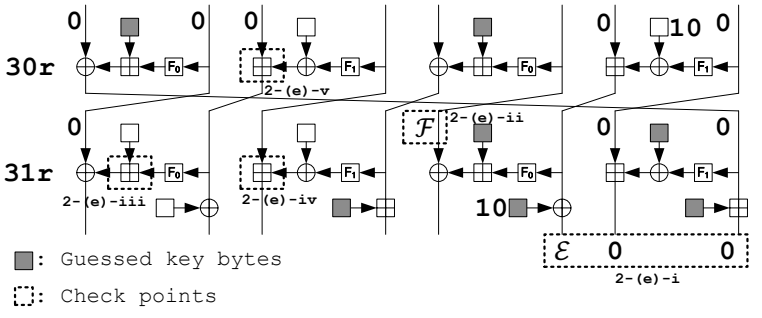


Fig. 3. Filtering of wrong quartets. The numbers nearby the check points indicate the corresponding steps from 2-(e)-i to 2-(e)-v.

Counting and sorting

3. In the following step 4 to step 12, discard the quartets with each key bytes guessed in each step which do not satisfy conditions. In each steps, if less than 3 quartets are remained, then go to Step 2 with another key guessing.
4. Guess the $LSB_4(K1[15])$ to compute $LSB_4(SK1[126])$ and its related keys. For each key and remained quartet, check (a) $LSB_4({}^{i}_{l}X_{29}[3] \oplus {}^{j}_{u}X'_{29}[3]) = 0$ and (b) $LSB_4({}^{i}_{v}X^{*}_{29}[3] \oplus {}^{j}_{w}X'^{*}_{29}[3]) = 0$.

5. Guess $MSB_4(K1[15])$ to compute $MSB_4(SK1[126])$ and its related keys. For each key and remained quartet, check (a) $MSB_4({}^{i}_{l}X_{29}[3] \oplus {}^{j}_{u}X'_{29}[3]) = 0$ and (b) $MSB_4({}^{i}_{v}X^{*}_{29}[3] \oplus {}^{j}_{w}X'^{*}_{29}[3]) = 0$.
6. Guess $LSB_4(K1[9])$ to compute $LSB_4(SK1[120])$ and its related keys. For each key and remained quartet, check (a) $LSB_4({}^{i}_{l}X_{30}[1] \oplus {}^{j}_{u}X'_{30}[1]) = 0$ and (b) $LSB_4({}^{i}_{v}X^{*}_{30}[1] \oplus {}^{j}_{w}X'^{*}_{30}[1]) = 0$.
7. Guess $MSB_4(K1[9])$ to compute $MSB_4(SK1[120])$ and its related keys. For each key and remained quartet, check (a) $MSB_4({}^{i}_{l}X_{30}[1] \oplus {}^{j}_{u}X'_{30}[1]) = 0$ and (b) $MSB_4({}^{i}_{v}X^{*}_{30}[1] \oplus {}^{j}_{w}X'^{*}_{30}[1]) = 0$.
8. Without key guessing, check if ${}^{i}_{l}X_{28}[0] \oplus {}^{j}_{u}X'_{28}[0] \in \mathcal{G}$ and ${}^{i}_{v}X^{*}_{28}[0] \oplus {}^{j}_{w}X'^{*}_{28}[0] \in \mathcal{G}$.
9. Without key guessing, check if ${}^{i}_{l}X_{28}[1] \oplus {}^{j}_{u}X'_{28}[1] = 0$ and ${}^{i}_{v}X^{*}_{28}[1] \oplus {}^{j}_{w}X'^{*}_{28}[1] = 0$.
10. Guess $LSB_4(K1[3])$ and $LSB_4(K1[11])$ to compute $LSB_4(SK1[122])$ and its related keys. For each key and remained quartet, check (a) $LSB_4({}^{i}_{l}X_{30}[5] \oplus {}^{j}_{u}X'_{30}[5]) = 0$ and (b) $LSB_4({}^{i}_{v}X^{*}_{30}[5] \oplus {}^{j}_{w}X'^{*}_{30}[5]) = 0$.
11. Guess $MSB_4(K1[3])$ and $MSB_4(K1[11])$ to compute $MSB_4(SK1[122])$ and its related keys. For each key and remained quartet, check (a) $MSB_4({}^{i}_{l}X_{30}[5] \oplus {}^{j}_{u}X'_{30}[5]) = 0$ and (b) $MSB_4({}^{i}_{v}X^{*}_{30}[5] \oplus {}^{j}_{w}X'^{*}_{30}[5]) = 0$.
12. Guess $K1[8]$ to compute $SK1[127]$ and its related keys. For each key and remained quartet, check (a) ${}^{i}_{l}X_{31}[7] \oplus {}^{j}_{u}X'_{31}[7] = 0$ and (b) ${}^{i}_{v}X^{*}_{31}[7] \oplus {}^{j}_{w}X'^{*}_{31}[7] = 0$.

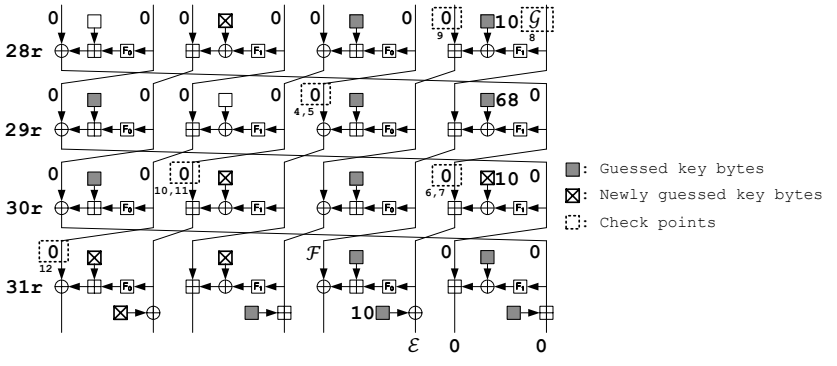


Fig. 4. Key counting procedure. The numbers nearby the check points indicate the corresponding steps from 4 to 12.

Searching with the list

13. If there exist a recorded $K1[0, 1, 2, 3, 5, 6, 8, 9, 10, 11, 12, 13, 14, 15]$ who have 3 or more remaining quartets $({}^{i}_{l}C, {}^{i}_{v}C^{*}, {}^{j}_{u}C', {}^{j}_{w}C'^{*})$, then exhaustively search the remaining two key bytes for $K1[4]$ and $K1[7]$ with more than two known plaintexts and its corresponding ciphertexts. If a 128-bit key is suggested, output it and its related keys as the keys of encryption oracles of the full rounds of HIGHT, otherwise go to Step 2 with another guess.

An overall view of our related-key rectangle attack is shown in Fig. 5 in Appendix C.

5.2 Complexity Analysis

The probability of 24-round related-key rectangle distinguisher used in our attack is $2^{117.68}$. So, we form the plaintext structures such that we are given $2^{119.68}$ quartets satisfying the input differences of the distinguisher and expect approximately 4 right quartets after guessing $K1[0, 1, 2, 5, 6, 10, 12, 13, 14]$.

Since we use structures S_i includes 2^{40} plaintexts and 16 kinds of input differences of distinguisher(set $\mathcal{A}$ which is defined in Section 4.2) is assumed, we have 2^{44} pairs per structure and we can consider 2^{2m+88} quartets from 2^m structures thus the number of required structures is $2^m = 2^{15.84}$. Hence, our attack requires $2^{15.84+40} = 2^{55.84}$ plaintexts and encrypts them with four encryption oracles defined by four related keys to get $2^{57.84}$ ciphertexts and that is the data complexity of our attack.

In the first step of the attack procedure in the above section, the number of queries to four encryption oracles with related keys is $2^{57.84}$ but this is negligible in total complexity.

We guess 9 bytes of $K1[0, 1, 2, 5, 6, 10, 12, 13, 14]$ in step 2 with a restriction on $K1[6]$ that $SK1[82] \in \mathcal{D}$, so total number of guessed key is 2^{70}. Moreover, an attack procedure and tested quartets when 9 bytes of K are guessed as $K1$ is identical to an attack procedure and tested quartets when 9 bytes of $K \boxplus \Delta^+ K$ are guessed as $K1$, so the total number of key guessing for $K1[0, 1, 2, 5, 6, 10, 12, 13, 14]$ is reduced to 2^{69}.

From step 2-(a) to 2-(d), we explain how we make $2^{119.68}$ quartets from $2^{15.84}$ structures and step (e), we explain how we filter out the wrong quartets. For steps from 2-(a) to 2-(d), we partially encrypt all plaintexts in each structure for 6/128 HIGHT with 4 related keys and choose the pairs satisfying the input differences of distinguisher, so these steps require

$$4 \times 6 \times 2^{-7} \times 2^{55.840+69} \approx 2^{122.425}$$

encryptions and yields two hash tables $\mathcal{H}$ and $\mathcal{I}$ of $2^{59.84}$ ciphertext pairs, respectively. In step 2-(e), we partially decrypt for 1/128 HIGHT and 2/128 HIGHT to compute $({}^{i}_{l}X_{31}[3] \oplus {}^{j}_{u}X'_{31}[3]$, ${}^{i}_{v}X^{*}_{31}[3] \oplus {}^{j}_{w}X'^{*}_{31}[3])$ and $(\Delta T, \Delta T')$, respectively, so these step requires

$$3 \times 2^{-7} \times 2^{57.840+69} \approx 2^{121.425}$$

decryptions.

In step 2-(e)-i, we check that two bytes of ciphertext differences are 0, and another one byte of ciphertext difference is equal to one of 8 elements in $\mathcal{E}$, so the filtering ratio of this step is $2^{-2\times2\times8-2\times5} = 2^{-42}$.

In step 2-(e)-ii, ${}^{i}_{l}X_{31}[3] \oplus {}^{j}_{u}X'_{31}[3]$ and ${}^{i}_{v}X^{*}_{31}[3] \oplus {}^{j}_{w}X'^{*}_{31}[3]$ must be one of 16 elements in $\mathcal{F}$, because $SK1[120] \oplus SK3[120] = SK2[120] \oplus SK4[120] =$ `0x10`, and $SK1[116] \oplus SK3[116] = SK2[116] \oplus SK4[116] =$ `0x68`, so the filtering ratio of this step is $2^{-2\times4} = 2^{-8}$.

Since the average ratio of Δx and Δz which satisfy that $\Pr[(\Delta x, 0) \xrightarrow{\boxplus} \Delta z] > 0$ is $2^{-2.82757}$, the filtering ratios of step 2-(e)-iii and iv are both $2^{-5.65514}$, and since

the average ratio of Δx, Δy, and Δz which satisfy that $\Pr[(\Delta x, \Delta y) \xrightarrow{\boxplus} \Delta z] > 0$ is $2^{-2.34852}$, the filtering ratio of step 2-(e)-v is $2^{-4.69704}$. Therefore, after filtering steps,

$$2^{119.68-42-8-2\times 5.65514-4.69704} = 2^{53.67268} < 2^{53.68}$$

quartets are left in average.

From step 3 to step 12 are key searching steps with $2^{53.68}$ quartets. The time complexities and number of remained quartets for each step are calculated in Table 3.

Table 3. Complexities of key searching steps

Step	Key guess (bit)	# of Quartets to test	Time Complexity	Elimination Ratio	# of remaining quartets	Key guess (sum,bit)	# of quartet per a key
4-(a)	4	$2^{53.68}$	$2^{1+53.68+4-7} = 2^{50.68}$	2^{-4}	$2^{53.68+4-4} = 2^{53.68}$	4	$2^{53.68-4} = 2^{49.68}$
4-(b)	0	$2^{53.68}$	$2^{1+53.68-7} = 2^{46.68}$	2^{-4}	$2^{53.68-4} = 2^{49.68}$	4	$2^{49.68-4} = 2^{45.68}$
5-(a)	4	$2^{49.68}$	$2^{1+49.68+4-8} = 2^{46.68}$	2^{-4}	$2^{49.68+4-4} = 2^{49.68}$	8	$2^{49.68-8} = 2^{41.68}$
5-(b)	0	$2^{49.68}$	$2^{1+49.68-8} = 2^{42.68}$	2^{-4}	$2^{49.68-4} = 2^{45.68}$	8	$2^{45.68-8} = 2^{37.68}$
6-(a)	4	$2^{45.68}$	$2^{1+45.68+4-7} = 2^{43.68}$	$2^{-1.5}$	$2^{45.68+4-1.5} = 2^{47.18}$	12	$2^{47.18-12} = 2^{35.18}$
6-(b)	0	$2^{47.18}$	$2^{1+47.18-7} = 2^{42.18}$	$2^{-1.5}$	$2^{47.18-1.5} = 2^{46.68}$	12	$2^{46.68-12} = 2^{34.68}$
7-(a)	4	$2^{46.68}$	$2^{1+46.68+4-7} = 2^{44.68}$	$2^{-1.5}$	$2^{46.68+4-1.5} = 2^{48.18}$	16	$2^{48.18-16} = 2^{32.18}$
7-(b)	0	$2^{48.18}$	$2^{1+48.18-7} = 2^{43.18}$	$2^{-1.5}$	$2^{48.18-1.5} = 2^{47.68}$	16	$2^{46.68-16} = 2^{30.68}$
8	0	$2^{47.68}$	$2^{2+47.68-7} = 2^{42.68}$	2^{-4}	$2^{47.68-4} = 2^{43.68}$	16	$2^{43.68-16} = 2^{27.68}$
9	0	$2^{43.68}$	$2^{2+43.68-7} = 2^{38.68}$	2^{-16}	$2^{43.68-16} = 2^{27.68}$	16	$2^{27.68-16} = 2^{11.68}$
10-(a)	8	$2^{27.68}$	$2^{1+27.68+8-7} = 2^{29.68}$	$2^{-2.58}$	$2^{27.68+8-2.58} = 2^{33.1}$	24	$2^{33.1-24} = 2^{9.1}$
10-(b)	0	$2^{33.1}$	$2^{1+33.1-7} = 2^{27.1}$	$2^{-2.58}$	$2^{33.1-2.58} = 2^{30.52}$	24	$2^{30.52-24} = 2^{6.52}$
11-(a)	8	$2^{30.52}$	$2^{1+30.52+8-7} = 2^{32.52}$	$2^{-2.58}$	$2^{30.52+8-2.58} = 2^{35.94}$	32	$2^{35.94-32} = 2^{3.94}$
11-(b)	0	$2^{35.94}$	$2^{1+35.94-7} = 2^{29.94}$	$2^{-2.58}$	$2^{35.94-2.58} = 2^{33.36}$	32	$2^{33.36-32} = 2^{1.36}$
12-(a)	8	$2^{33.36}$	$2^{1+33.36+8-7} = 2^{35.36}$	$2^{-5.17}$	$2^{33.36+8-5.17} = 2^{36.19}$	40	$2^{35.19-40} = 2^{-4.81}$
12-(b)	0	$2^{36.19}$	$2^{1+35.19-7} = 2^{30.19}$	$2^{-5.17}$	$2^{36.19-5.17} = 2^{31.02}$	40	$2^{31.02-40} = 2^{-8.98}$
Total	40		$2^{50.9001}$		$2^{30.74}$	40	$2^{31.02-40} = 2^{-8.98}$

Time complexities in Table 3 except step 8 and step 9 are calculated by the early abort technique [11] so these steps are divided into two sub-steps which check that each pair of ciphertexts or intermediate values is valid for a right quartet. The time complexities for each steps are calculated by the multiplications of the number of partially decrypted ciphertexts, the number of remaining quartets from the previous step, the number of guessed keys, and the ratio of partial rounds to HIGHT encryption.

In steps 6, 7, 10, 11, and 12, although we check 8-bit difference. Since a part of quartets are already discarded in the filtering steps by some conditions for the 8-bit difference, we can eliminate with the remaining ratios.

The number of remaining quartets is calculated by multiplication of the number of remaining quartets from the previous step, the number of guessed keys in current step, and the elimination ratio of current step. Using this, we check that how many quartets are counted for each guessed key in average, and if this ratio is significantly less than 1, we can conclude that the right quartets and right key are distinguished from wrong quartets and wrong keys. After step 12, total

number of guessed keys is 2^{40} and the number of remaining quartets is $2^{31.02}$, so we expect

$$2^{31.02-40} = 2^{-8.98}$$

quartets are remained for a key in average while more than 3 quartets are remained if guessed key is right key. Thus we have to test quartets until step 12, and the computational complexity from step 4 to step 12 is $2^{50.9001}$.

Therefore, the computational complexity of our related-key rectangle attack for a quarter of key space is

$$2^{122.425} + 2^{121.425} + 2^{69+50.9001} < 2^{123.169},$$

and since the computational complexity of exhaustive key searching for the remaining part of key space is $3 \times 2^{124} = 2^{125.585}$, the total computational complexity of our related-key attack is

$$2^{123.169} + 2^{125.585} = 2^{125.833}.$$

6 Conclusions

In this paper, we find a 24-round related-key rectangle distinguisher using a local-collision property of 4 rounds of HIGHT and extremely deep ladder switch technique when add-differences are used for a relations of keys. This distinguisher can be regards as a 25-round distinguisher because the distinguisher is followed by one round truncated differential trail with probability 1. Based on this distinguisher, we present a related-key rectangle attack on the full rounds of HIGHT for a large weak key space and we consider a related-key attack which is valid for whole key space faster than 2^{126} encryptions required for the exhaustive key searching with 4 related keys. Time complexity of our attack is very marginal and seems to be hard to realize to extract the secret key bits. However, our result gives an evidence for the fact that HIGHT cannot be regarded as an instantiation of the ideal cipher.

References

1. Biham, E.: How to Forge DES-Enhanced Messages in 2^{28} Steps. CS 884 (August 1996)
2. Biham, E., Dunkelman, O., Keller, N.: A Related-Key Rectangle Attack on the Full KASUMI. In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 443–461. Springer, Heidelberg (2005)
3. Biryukov, A., Dunkelman, O., Keller, N., Khovratovich, D., Shamir, A.: Key Recovery Attacks of Practical Complexity on AES Variants With Up To 10 Rounds. To appear in EUROCRYPT 2010, Available at Cryptology ePrint Archive, Report 2009/374 (2010), `http://eprint.iacr.org/2009/374`
4. Biryukov, A., Khovratovich, D.: Related-key cryptanalysis of the full AES-192 and AES-256. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 1–18. Springer, Heidelberg (2009)

5. Biryukov, A., Khovratovich, D.: Feasible Attack on the 13-round AES-256. Cryptology ePrint Archive, Report 2010/257
6. Chabaud, F., Joux, A.: Differential Collisions in SHA-0. In: Krawczyk, H. (ed.) CRYPTO 1998. LNCS, vol. 1462, pp. 56–71. Springer, Heidelberg (1998)
7. Dunkelman, O., Keller, N., Shamir, A.: A Practical-Time Related-Key Attack on the KASUMI Cryptosystem Used in GSM and 3G Telephony. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 393–410. Springer, Heidelberg (2010)
8. Hong, D., Sung, J., Hong, S., Kim, J., Lee, S., Koo, B.-S., Lee, C., Chang, D., Lee, J., Jeong, K., Kim, H., Kim, J., Chee, S.: HIGHT: A New Block Cipher Suitable for Low-Resource Device. In: Goubin, L., Matsui, M. (eds.) CHES 2006. LNCS, vol. 4249, pp. 46–59. Springer, Heidelberg (2006)
9. International Organization for Standardization. ISO/IEC 18033-3:2005. Information technology – Security techniques – Encryption algorithms – Part 3: Block ciphers (2005)
10. Lu, J.: Cryptanalysis of reduced versions of the HIGHT block cipher from CHES 2006. In: Nam, K., Rhee, K. (eds.) ICISC 2007. LNCS, vol. 4817, pp. 11–26. Springer, Heidelberg (2007)
11. Lu, J.: Cryptanalysis of Block Ciphers. PhD thesis, Royal Holloway, University of London, England (July 2008)
12. Lipmaa, H., Moriai, S.: Efficient Algorithms for Computing Differential Properties of Addition. In: Matsui, M. (ed.) FSE 2001. LNCS, vol. 2355, pp. 336–350. Springer, Heidelberg (2002)
13. Özen, O., Varıcı, K., Tezcan, C., Kocair, Ç.: Lightweight block ciphers revisited: Cryptanalysis of reduced round PRESENT and HIGHT. In: Boyd, C., González Nieto, J. (eds.) ACISP 2009. LNCS, vol. 5594, pp. 90–107. Springer, Heidelberg (2009)
14. Vaudenay, S.: When is an Algorithm Legally Broken? Early Symmetric Crypto (ESC) Seminar (January 14, 2010)
15. Zhang, P., Sun, B., Li, C.: Saturation Attack on the Block Cipher HIGHT. In: Garay, J.A., Miyaji, A., Otsuka, A. (eds.) CANS 2009. LNCS, vol. 5888, pp. 76–86. Springer, Heidelberg (2009)

A Some Flaws in Previous Attack on Reduced Rounds of HIGHT

As mentioned in Section 1, Lu et al. present a related-key rectangle attack on 26 rounds of HIGHT which uses two related-key differential trails for 10 rounds(for $E0$) and 8 rounds(for $E1$) of HIGHT, respectively. Their 10-round related-key differential trail for $E0$ covers rounds from 3 to 12, with the following input and output differences,

$$(\texttt{0x2a}, \texttt{0x43}, \texttt{0x80}, \texttt{0x0}, \texttt{0x0}, \texttt{0x0}, \texttt{0x0}, \texttt{0x0}) \longrightarrow (\texttt{0x0}, ?, ?, \texttt{0x80}, \texttt{0x0}, \texttt{0x0}, \texttt{0x0}, \texttt{0x0}),$$

where the relation of the key is $\Delta K[2] = \texttt{0x80}$ and $\Delta K[i] = \texttt{0x0}$ for $i = 0, 1, 3, 4, 5, ..., 15$. They compute the amplified probability $2^{-19.98}$ of $E0$ for some possible values for positions marked by '?', and this probability is computed based on the fact that the probability of the first 1-round differential trail,

$$(\texttt{0x2a}, \texttt{0x43}, \texttt{0x80}, \texttt{0x0}, \texttt{0x0}, ..., \texttt{0x0}) \rightarrow (\texttt{0x43}, \texttt{0x80}, \texttt{0x0}, \texttt{0x0}, ..., \texttt{0x0}, \texttt{0x0})$$

is 2^{-3}. This probability arise from the following equation (4).

$$\Pr_{x,k}[(x \boxplus k) \oplus ((x \oplus \texttt{0x2a}) \boxplus k) = \texttt{0x2a}] = 2^{-3}. \tag{4}$$

Here, we can observe that for a fixed k, the probability in equation (4) can be different from 2^{-3}, moreover, the probability in equation (4) is 0 for 148 out of 256 keys. Also, the probability of the related-key differential trail for $E1$ has the same flaws. The number of k such that the probability of the first round of $E1$ is 0 is 158.

In block cipher cryptanalysis, a target secret key is assumed to be fixed, so in some cases the related-key differential trails for $E0$ and $E1$ in [10] are not satisfied with suggested probabilities. Thus, the related-key rectangle attack in [10] is regarded as an attack valid only for weak keys.

B Exhaustive Key Searching in the Related-Key Model

The validity of an attack on a cipher is usually proved through comparison of the complexities with those of an exhaustive key searching in the same attack model. Let $f_1, ..., f_{t-1}$ be simple bijective relations. We assume that we are given t distinct encryption oracles $E_{f_0(K)}$, $E_{f_1(K)}$, ..., $E_{f_{t-1}(K)}$ for a related-key tuple $(f_0(K), f_1(K), ..., f_{t-1}(K))$, where f_0 is the identity function. We also assume that the encryption oracle has the block size n and the key space $\mathcal{K}$ has 2^k elements. Especially, we focus on the case of $k/2 \leq n < k$ because our target is HIGHT. In this setting, the exhaustive key searching consists of the following phase.

1. Choose and fix two plaintexts P and P^*, and get the ciphertexts C_i and C_i^* for each encryption oracle $E_{f_i(K)}$.
2. Repeat the following phases.
   (a) Randomly pick one K' of key candidates, compute $C = E_{K'}(P)$, and check whether there exists a C_i such that $C = C_i$.
   (b) If such C_i is found, compute $C^* = E_{K'}(P^*)$, and check whether $C_i^* = C^*$.
   (c) If a match $(C, C^*) = (C_i, C_i^*)$ is found, halt and output K' as the right value of $f_i(K)$. Otherwise, discard $K', f_1^{-1}(K'), ..., f_{t-1}^{-1}(K')$ from search space, and go to (a) and try again.

This is not new and similar approaches were mentioned in [1,14]. We can expect a match is found with $2^{k-\log_2 t}$ trials. The match yields one of $f_0(K), f_1(K), ..., f_{t-1}(K)$ with a high probability. So, the time complexity of this attack is dominated by $2^{k-\log_2 t}$ encryptions. Therefore our attack is forced to have the time complexity less than 2^{126}, since we use $t = 4$ related keys of k =128 bits.

C An Overall View of Our Related-Key Rectangle Attack

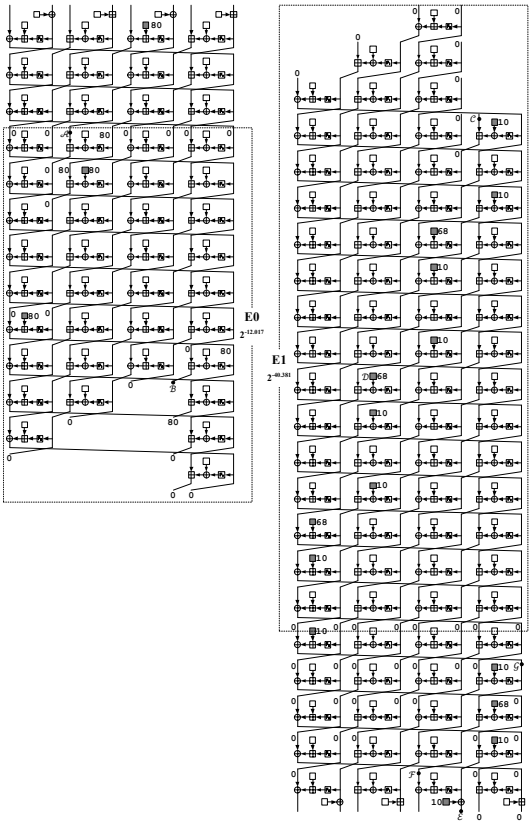


Fig. 5. Related-key differential trails of HIGHT