SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

January 2020, Vol. 63 112106:1–112106:10
https://doi.org/10.1007/s11432-019-1507-1

Cryptanalysis of PRIMATEs

Yanbin LI[1], Meiqin WANG[1], Wenqing LIU[1] & Wei WANG[1,2*]

[1] *Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan* 250100*, China;*
[2] *Shandong Provincial Key Laboratory of Computer Networks, Jinan* 250100*, China*

Received 16 March 2019/Revised 16 May 2019/Accepted 2 August 2019/Published online 24 December 2019

Abstract PRIMATEs is a family of authenticated encryption design submitted to competition for authenticated encryption: security, applicability, and robustness. The three modes of operation in PRIMATEs family are: APE, HANUMAN, GIBBON with security levels: 80, 120 bits. APE is robust despite the nonce misusing. In this study, we revise the algebraic model and find new integral distinguishers for both PRIMATE permutation and its inverse permutation. Moreover, we construct a zero-sum distinguisher for full 12-round PRIMATE-80/120 permutation with the $2^{100}/2^{105}$ complexity, improving over previous work. We also perform an integral attack on 8-round finalization of APE-80/120 with 2^{30} chosen messages. The key recovery process is optimized using the FFT technique presented by Todo and Aoki. Our work is the best attack against APE, demonstrating the practical attack on 8-round finalization of APE-80. The new integral distinguishers apply to create forgeries on 5/6-round finalization of APE and HANUMAN that require $2^{15}/2^{30}$ chosen messages, which is the first forgery attack against APE and HANUMAN.

Keywords PRIMATEs, APE, HANUMAN, integral distinguisher, key recovery attack, forgery

Citation Li Y B, Wang M Q, Liu W Q, et al. Cryptanalysis of PRIMATEs. Sci China Inf Sci, 2020, 63(1): 112106, https://doi.org/10.1007/s11432-019-1507-1

1 Introduction

The rapid development of the Internet of things (IoT) has facilitated the application environments of authenticated encryption (AE). Rogaway [1] formalizes the AE concept to guarantee data confidentiality and authenticity using an encryption scheme and a message authentication code (MAC). Such generic composition paradigm is discussed by [2]. However, a crude combination of the two schemes can cause serious problems, such as poor performance and security evaluation problem. To overcome these problems, Jutla [3], Gligor et al. [4], and Rogaway et al. [5] proposed an integrated AE scheme that provides confidentiality and authenticity in a single scheme. Presently, AES-GCM [6, 7] is one of the most widely deployed AE schemes defined in NIST's SP 800-38D and P1619.

In 2013, NIST sponsored competition for authenticated encryption: security, applicability, and robustness (CAESAR) [8] to find suitable AE schemes for various environments. In CAESAR competition, the submission protfolio is organized into three use cases: (1) lightweight applications, (2) high-performance applications, and (3) defense in depth. On March 2014, PRIMATEs competed in the CAESAR competition and passed the second-round filtering on the September in the same year. Evaluating the security of submissions promotes the process of CAESAR competition.

Andreeva et al. [9] designed PRIMATEs, the designers slightly modified the first version of PRIMATEs v1.0 architecture to upgrade it to PRIMATEs v1.02. Three modes of operation of AE schemes family

[*] Corresponding author (email: weiwangsdu@sdu.edu.cn)

PRIMATEs are: APE, HANUMAN, GIBBON with two security levels: 80 and 120 bits. The primary recommended security level for these versions is 120. APE is the primary recommended mode followed by HANUMAN and GIBBON. APE is robust despite misusing the nonce, whereas HANUMAN and GIBBON are secure if the nonce is unique and non-repeating. In the v1.02, the designers submitted a document that revealed the first cryptanalysis of PRIMATEs, including differential trails, linear trails, collision trails, and impossible differential trails for the PRIMATE permutation. Saha et al. [10] also performed a classical diagonal fault attack on APE. Minaud presented key-recovery attacks on 8-round finalization of APE with 2^{33} chosen messages, then pointed out that the full 12-round PRIMATE permutation can be distinguished from a perfect random permutation using a zero-sum distinguisher with the complexity 2^{130} [11]. Morawiecki et al. [12] applied a cube-like attack on 6-round initialization of HANUMAN-120 in nonce-respecting scenario. Lukas and Daemen [13] performed a state recovery attack on GIBBON in nonce-reuse scenario based on a flaw of the 6-round PRIMATE permutation.

1.1 Contributions

We improve the previous cryptanalysis work of APE and HANUMAN in PRIMATEs family, including zero-sum distinguisher on PRIMATE permutation, key-recovery attacks on APE, and forgery attacks on APE and HANUMAN. The results of the PRIMATEs authenticated encryption are summarized in Table 1. We provide the following contributions.

1.1.1 *Improved integral distinguishers*

We provide a more precise evaluation of the algebraic degree of PRIMATE permutation and we find new integral distinguishers for both forward and backward rounds of PRIMATE permutation useful for many attacks.

1.1.2 *Improved zero-sum distinguishers*

Minaud [11] introduced zero-sum distinguishers for 12-round PRIMATE-80/120 permutation by the complexity 2^{130} considering the algebraic degree of the permutation. We combine two integral distinguishers (targeting PRIMATE permutation and its inverse) to introduce new zero-sum distinguishers into full 12-round PRIMATE-80/120 permutation with the reduced complexity $2^{100}/2^{105}$.

1.1.3 *Improved key-recovery attacks on APE*

Minaud [11] performed a cube attack against 8-round finalization of APE-80/120, which required 2^{33} chosen 2-block messages and time complexity $2^{61}/2^{71}$. We apply new integral distinguishers and the fast fourier transform (FFT) technique [14] to perform key-recovery attacks on APE-80/120 with 2^{30} chosen 1-block messages and substantially reduced time complexity $2^{39.29}/2^{50.26}$. We performed the practical attack against 8-round finalization of APE-80 for the first time based on the complexity. APE is robust despite misusing the nonce. Therefore, we evaluate its security in a nonce-misuse scenario.

1.1.4 *Forgery attacks on APE and HANUMAN*

From our discovery, the new integral distinguishers we can be applied to the 5/6-round finalization of APE and HANUMAN to create forgery with practical complexity $2^{15}/2^{30}$ in the nonce-misuse scenario. We construct a structure that contain m messages, with knowing last blocks of ciphertexts and tags for $(m-1)$ messages. This procedure determines the last block of ciphertext and tag for the remaining message which threatens the integrity of reduced-round APE. Since HANUMAN demands unique nonces, we compromise the integrity of its variant to repeat nonces.

1.2 Organization

This paper is organized as follows. In Section 2, we show notations used in this paper and a brief description of PRIMATEs. Section 3 provides new integral distinguishers for both PRIMATE permutation and

Table 1 Results for PRIMATEs

Attack type	Variants	Rounds	Data complexity	Time complexity	Method	Scenario	Source
Distinguisher	PRIMATE-80	12/12	–	2^{130}	Zero-sum	–	Ref. [11]
		12/12	–	2^{100}			Section 4
	PRIMATE-120	12/12	–	2^{130}			Ref. [11]
		12/12	–	2^{105}			Section 4
Key recovery	APE-80	8/12	2^{33}	2^{61}	Cube	Nonce-misuse	Ref. [11]*
		8/12	2^{35}	2^{61}	Cube		Ref. [11]
		8/12	2^{35}	$2^{39.29}$	Integral	Nonce-misuse	Section 5
		8/12	2^{30}	$2^{39.29}$	Integral		Section 5
Key recovery	APE-120	8/12	2^{33}	2^{71}	Cube	Nonce-misuse	Ref. [11]*
		8/12	2^{35}	2^{71}	Cube		Ref. [11]
		8/12	2^{35}	$2^{50.26}$	Integral	Nonce-misuse	Section 5
		8/12	2^{30}	$2^{50.26}$	Integral		Section 5
Forgery	APE-80	5/12	2^{15}	2^{15}	Integral	Nonce-misuse	Section 6
	APE-80	6/12	2^{30}	2^{30}			Section 6
	APE-120	5/12	2^{15}	2^{15}			Section 6
	APE-120	6/12	2^{30}	2^{30}			Section 6
Key recovery	HANUMAN-120	6/12	2^{65}	2^{65}	Cube-like	Nonce-respecting	Ref. [12]
Forgery	HANUMAN-80	5/12	2^{15}	2^{15}	Integral	Nonce-misuse	Section 6
	HANUMAN-80	6/12	2^{30}	2^{30}			Section 6
	HANUMAN-120	5/12	2^{15}	2^{15}			Section 6
	HANUMAN-120	6/12	2^{30}	2^{30}			Section 6

* The attack required 2-block message encryption instead of 1-block messages, with the first block fixed arbitrarily, and the second block covering a cube of size 33.

Table 2 Notations

Symbol	Definition
$x \in \{0,1\}^k$	Bitstring x of length k (variable if $k = *$)
$x \oplus y$	XOR of bitstrings x and y
$x\|\|y$	Concatenation of bitstrings x and y
x_i	i-th bit of the state used in PRIMATE permutation
X_i	5-bit state word of the state used in PRIMATE permutation
K, N, T	Secret key K, nonce N, tag T
P, C, A	Plaintext P, ciphertext C, associated data A (in blocks P_i, C_i, A_i)
p_1, p_2, p_3, p_4	Four different permutations used in PRIMATEs

its inverse permutation. In Section 4, we apply the new integral distinguishers to present a new zero-sum distinguisher on 12-round PRIMATE-80/120 permutation. In Section 5, we perform key-recovery attacks on round-reduced version of APE. We create forgeries for round-reduced version of APE and HANUMAN with 5/6-round finalization in Section 6. Section 7 concludes the paper.

2 Preliminary

In this section, we show notations and provide a brief description of AE family PRIMATEs.

2.1 Notations

Table 2 specifies the notations in this paper.

2.2 Brief description of PRIMATEs [9]

PRIMATEs is a family of AE schemes designed by Andreeva et al. [9] and defined by three modes of operation Scheme $\in \{\mathrm{APE},\ \mathrm{HANUMAN},\ \mathrm{GIBBON}\}$, it comprises the two security levels $s \in \{80, 120\}$

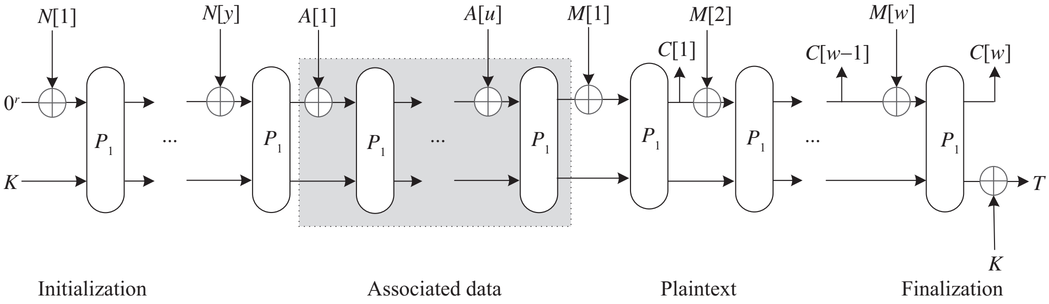

Figure 1 The encryption of APE.

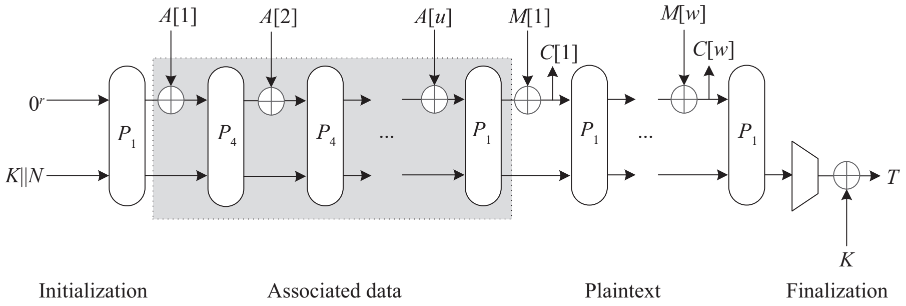

Figure 2 The encryption of HANUMAN.

Table 3 The S-box of PRIMATEs [9]

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$S(x)$	1	0	25	26	17	29	21	27	20	5	4	23	14	18	2	28
x	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
$S(x)$	15	8	6	3	13	7	24	16	30	9	31	10	22	12	11	19

bits. Scheme is based on duplex sponge construction. The underlying permutation of PRIMATEs is called PRIMATE-s which operates on a sponge state of size b bits, consisting of a rate part with r bits and a capacity part with c bits. PRIMATEs has two different sizes; each size has 4 variants (p_1, p_2, p_3, p_4). Each mode of operation and security level determines the length of the key, tag, nonce, and the specific permutation in PRIMATE-s. Ref. [9] provides the details.

We focus on APE and HANUMAN, which are shown in Figures 1 and 2. The encryption is partitioned into four phases: initialization, processing associated data, processing the plaintext, and finalization.

PRIMATE permutation is inspired by a wide trail strategy [15] and its structure is similar to that of Rijndael [16]. PRIMATE-80 and PRIMATE-120 operate on a 5 × 8 and a 7 × 8 state of 5-bit words, respectively. The first row of the state (5 bytes) is the rate part, whereas the others are the capacity part of the state. PRIMATE updates the state in four steps:

$$\mathrm{CA} \circ \mathrm{MC} \circ \mathrm{SR} \circ \mathrm{SE}.$$

Permutations p_1, p_2, p_3, and p_4 of PRIMATE differ in the number of rounds and round constant used in CA step.

2.2.1 *SubElements* (*SE*)

Table 3 defines the 5-bit S-box in the SubElements step, applied to each parallel word of the state. This step is the only non-linear operation in PRIMATE permutation.

2.2.2 *ShiftRows* (*SR*)

The ShiftRows step is a left cyclic shift operation based on the words of the state row by row. Row i is shifted left by $s_i = \{0, 1, 2, 4, 7\}$ positions for PRIMATE-80 and by $s_i = \{0, 1, 2, 3, 4, 5, 7\}$ positions for

PRIMATE-120.

2.2.3 *MixColumns* (*MC*)

The MixColumns step is on the state column by column. It is a left-multiplication by a 5 × 5 matrix for PRIMATE-80 and a 7 × 7 matrix for PRIMATE-120.

2.2.4 *ConstantAddition* (*CA*)

The ConstantAddition step XORed the second word of the second row with a constant.

3 Integral distinguisher

In this section, we revise the algebraic model of PRIMATE permutation and give a more accurate estimate on the algebraic degree. Then we find new integral distinguishers for both forward and backward rounds of PRIMATE permutation for subsequent work.

3.1 Algebraic model of PRIMATE permutation

To find the integral distinguishers for both PRIMATE permutation and its inverse, we must determine the degree of one PRIMATE S-box and its inverse.

Lemma 1. The algebraic degree of one PRIMATE permutation round is 2.

Proof. With respect to F_2, the algebraic degree of one PRIMATE S-box can be easily determined from its algebraic normal form (ANF):

$$\begin{aligned}
y_0 &= x_0x_2 + x_0x_3 + x_1x_4 + x_1 + x_2x_3 + x_2 + x_3,\\
y_1 &= x_0 + x_1x_2 + x_1x_3 + x_2x_3 + x_2x_4 + x_3,\\
y_2 &= x_0x_1 + x_0x_4 + x_0 + x_1 + x_2x_3 + x_2x_4,\\
y_3 &= x_0x_2 + x_0x_4 + x_0 + x_1x_2 + x_4x_3,\\
y_4 &= x_0x_3 + x_1 + x_2x_4 + x_4 + 1.
\end{aligned}$$

Here, x_0, x_1, x_2, x_3, x_4, and y_0, y_1, y_2, y_3, y_4 represent the input, and output of an S-box, with x_0/y_0 representing the most significant bit. The S-boxes in one SE step are applied in parallel to the state. Moreover, SR, MC and CA do not increase the algebraic degree. Consequently, the overall degree of one PRIMATE permutation round is 2.

Lemma 2. The algebraic degree of one inverse PRIMATE permutation round is 3.

Proof. We use the ANF of the inverse PRIMATE S-box to determine the degree of the inverse permutation:

$$\begin{aligned}
y_0 = {} & x_0x_1 + x_0x_2x_3 + x_0x_2x_4 + x_0x_2 + x_0x_3x_4 + x_0x_3\\
& + x_0x_4 + x_0 + x_1x_3x_4 + x_1 + x_2x_3x_4 + x_2x_3 + x_3x_4,\\
y_1 = {} & x_0x_1x_3 + x_0x_1x_4 + x_0x_2x_3 + x_0x_2x_4 + x_0x_3x_4 + x_1x_2x_3\\
& + x_1x_4 + x_2x_3x_4 + x_2 + x_3x_4 + x_3,\\
y_2 = {} & x_0x_1x_4 + x_0x_2x_4 + x_0x_2 + x_0x_3x_4 + x_0x_3 + x_0 + x_1x_2x_3\\
& + x_1x_2 + x_1x_3 + x_2x_3 + x_3x_4 + x_3,\\
y_3 = {} & x_0x_1x_3 + x_0x_1x_4 + x_0x_2x_4 + x_0x_4 + x_0 + x_1x_2x_4\\
& + x_1x_2 + x_2x_3 + x_2x_4 + x_2 + x_3,\\
y_4 = {} & x_0x_1x_2 + x_0x_1 + x_0x_2x_3 + x_0x_2x_4 + x_0x_3x_4 + x_0x_3\\
& + x_1x_2x_4 + x_1x_2 + x_1x_3x_4 + x_1x_3 + x_1x_4 + x_2x_3 + x_2\\
& + x_3 + x_4 + 1.
\end{aligned}$$

The algebraic degree of the ANF of the inverse PRIMATE S-box is 3. Similar to Lemma 1, the overall degree of one inverse PRIMATE permutation round is 3.

3.2 Integral distinguishers for forward/backward rounds of PRIMATE permutation

According to Lemma 1, an upper bound for the degree of the r-round PRIMATE-80/120 permutation is 2^r, showing that an r-round integral distinguisher requires $d = 2^r + 1$ bits to take over all possible values. However, we can still improve by setting d a multiple of the 5-bit S-box size [17]. By choosing d active bits always including complete S-boxes, the inputs (and consequently outputs) of these S-boxes will loop through all possible values with the remain S-boxes constant inputs (and consequently outputs). When we focus on the output of SE step, some there are still d bits to be active and the remain to be fixed. Thus, an additional round is added to the integral distinguisher with the same data.

As an illustration, a 5/6/7-round integral distinguisher for PRIMATE permutation is constructed with $2^{20}/2^{35}/2^{65}$.

The technique to build the integral distinguisher for the inverse PRIMATE permutation is similar. According to Lemma 2, an r backward rounds integral distinguisher needs $d = 3^r + 1$ bits to loop through all possible values. If we use the active bits to cover complete S-boxes, we can add one round to improve the existing distinguisher.

For example, a 5-round integral distinguisher for the inverse PRIMATE permutation can be constructed with 2^{85}.

3.3 Improved integral distinguishers for forward rounds of PRIMATE permutation

The major problem herein is to find a precise evaluation for the algebraic degree of PRIMATE permutation after several rounds. Yang and Lai [18] computed the algebraic degree of n-variable boolean function f without the knowledge of ANF of this function. If the degree of the function f is less than d, the relation $\sum_{i=0}^{2^d-1} f(x \oplus i) = 0$ holds for some x. When $\sum_{i=0}^{2^d-1} f(x \oplus i) = 0$ holds for all possible x, the algebraic degree of the function f is less than d. However, this is not practical for permutations with very big state. However, we only need to check whether $\sum_{i=0}^{2^d-1} f(x \oplus i) = 0$ for $\lceil 1.0294(n+1) \rceil$ random values of x to determine the algebraic degree of f [18].

$n = 200/280$ for PRIMATE-80/120. We slightly modify Algorithm 4 in [18] to Algorithms 1 and 2 to obtain the algebraic degree only regarding to $x_0, x_1, \ldots, x_{14}$ or $x_0, x_1, \ldots, x_{29}$ of 5-round or 6-round PRIMATE-80/120 permutation, respectively. We use general notation f to denote the 5/6-round PRIMATE permutation and x to denote state bits. If different bits are needed to active, we must modify the i parameter in Algorithms 1 and 2.

Algorithm 1 Distinguisher searching algorithm for 5-round PRIMATE permutation

```
for $0 < s < \lceil 1.0294(n+1) \rceil$ do
    Pick $x$ randomly;
    Compute tmp $= \sum_{i=0}^{2^{15}-1} f(x \oplus i)$;
    if tmp $\neq 0$ then
        Output "Proposition 1 does not hold";
    end if
end for
```

Algorithm 2 Distinguisher searching algorithm for 6-round PRIMATE permutation

```
for $0 < s < \lceil 1.0294(n+1) \rceil$ do
    Pick $x$ randomly;
    Compute tmp $= \sum_{i=0}^{2^{30}-1} f(x \oplus i)$;
    if tmp $\neq 0$ then
        Output "Proposition 2 does not hold";
    end if
end for
```

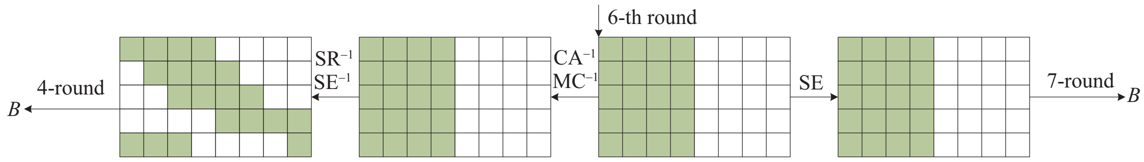

Figure 3 (Color online) Zero-sum distinguisher for PRIMATE-80.

The searching leads to Propositions 1 and 2.

Proposition 1. If we choose the first 3 S-boxes to loop through all possible values and other bits are constants, a balance is reached after 5-round PRIMATE-80/120 permutation.

Proposition 2. If we choose the first 6 S-boxes to loop through all possible values and other bits are constants, a balance is reached after 6-round PRIMATE-80/120 permutation.

4 Improved zero-sum distinguishers

Aumasson and Meier [19] presented a new distinguishing property, named the zero-sum property to show a non-ideal property of functions. The distinguisher based on the zero-sum property can search a set of inputs of a given function, which sum to zero, and corresponding outputs which also sum to zero over F_2. This is built by starting from the middle of the function and combining two integral distinguishers for forward and backward rounds. Minaud [11] found zero-sum distinguishers of the 12-round PRIMATE-80/120 permutation with the complexity 2^{130}. In this section, we use the integral distinguishers in the previous section to present improved zero-sum distinguishers for 12-round PRIMATE-80/120 permutation with the complexity $2^{100}/2^{105}$.

4.1 Zero-sum distinguisher for PRIMATE-80

For PRIMATE-80 permutation, the intermediate state of the input of the 6-th round is at the junction point. We set the number of active bits d to be a multiple of the 25-bit column size. In other words, we choose complete columns including complete S-boxes to be active and fix the remaining constants. In the forward direction, we add an additional round or reduce the required data. In the backward direction, the complete active columns are unaffected by MC step as this step is linear and operating on the state by column. Thus, active complete columns would be still active after an inverse MC step. Then, active complete S-boxes covered in active columns allow the distinguisher to go through the SE step with no cost in the 5-th round.

In this case, we attack 7 forward rounds and 5 backward rounds. Now, we choose 20 S-boxes in the first 4 columns and make the corresponding input bits for the 20 S-boxes loop through all possible values. The remaining bits are set to constant. Then, the zero-sum distinguisher for PRIMATE-80 is formalized as Proposition 3. As shown in Figure 3, green words denote active bits; white words are set to constants; and B means that the sum of input/output states is zero.

Proposition 3. For the full 12-round PRIMATE-80 permutation, the four columns of the input of the 6-th round take over all possible values and the others are set to constants, all inputs of the 12-round permutation XOR to zero, and all corresponding outputs of the 12-round permutation also XOR to zero.

The zero-sum distinguisher is built in the following way:

(1) In the forward direction, a 7-round integral distinguisher requires at least the data complexity 2^{65}.

(2) In the backward direction, a 5-round integral distinguisher requires at least the data complexity 2^{85}.

(3) In all conditions, we prepare a set of $2^{5\times 20} = 2^{100}$ bits including the complete four columns of the input state of 6-th round to take over all possible values, in which the remains are set to constant. To build a zero-sum distinguisher for PRIMATE-8, the active set propagates 7 forward rounds and 5 backward rounds.

4.2 Zero-sum distinguisher for PRIMATE-120

For PRIMATE-120, Proposition 4 is used to describe the zero-sum distinguisher with the complexity 2^{105}.

Proposition 4. For the full 12-round PRIMATE-120 permutation, the 3 columns of the 6-th round input take over all possible values, whereas the others are constants, all inputs of the 12-round permutation XOR to zero and all corresponding outputs of the 12-round permutation also XOR to zero.

The 12-round zero-sum distinguisher for PRIMATE-120 and PRIMATE-80 is similar. We start from the intermediate state of the input of the 6-th round. We attack a 7 forward rounds and a 5 backward rounds, in which their integral distinguishers need at least the data complexity 2^{65} and 2^{85}, respectively. Thus, we choose 21 S-boxes in the three complete columns and make the corresponding input bits for these 21 S-boxes loop through all possible values. The remaining bits are set to constants. The two integral distinguishers are combined to construct a 12-round zero-sum distinguisher for PRIMATE-120 permutation with the complexity 2^{105}.

For a perfect random 200/280-bit permutation, we choose a set of the inputs and corresponding outputs, the probability of the outputs add up to zero over F_2 is $2^{-200}/2^{-280}$ when the XOR of the corresponding inputs is zero. Thus, PRIMATE-80/120 permutation could be certainly distinguished from the random permutation.

5 Improved key-recovery attacks on APE

At CANS'14, Todo and Aoki [14] presented the FFT key recovery technique for the integral attack to reduce the time complexity. When the integral distinguisher uses N chosen plaintexts and the guessed key has k bits, a straightforward key recovery requires the time complexity $O(N2^k)$. However, the FFT key recovery method requires only the time complexity $O(N + k2^k)$. The calculation method uses fast Walsh-Hadamard transform (FWHT) instead of the FFT, which requires a total time complexity of approximately $4k2^k$ additions. We apply the FFT key recovery technique to recover the key.

We use the integral distinguisher given in Subsections 3.2 and 3.3 for 6-round PRIMATE-80/120 permutation, in order to perform key-recovery attacks on 8-round finalization of APE-80/120 with data complexity 2^{30} and time complexity $2^{39.29}/2^{50.26}$, respectively.

5.1 Integral attack on APE-80

For APE-80, the first row of the intermediate state is observable if more than one block of plaintext is need to be processed before the phase of finalization. If we arbitrarily choose 7 state words to loop through all possible values while the others are set to constants, the corresponding outputs of 6-round PRIMATE-80 permutation add up to zero over F_2. The detail of this distinguisher is described in Subsection 3.2.

As illustrated in Figure 4, we use the 6-round integral distinguisher to perform a key-recovery attack on 8-round finalization of APE-80. The gray words are known, whereas the white words are unknown, the orange words in the equivalent key eK (eK $= \mathrm{MC}^{-1}(0^{40}||K)$) denote the guessed key words in the attack. thus, we guess 25-bit equivalent key in the 8-th round and decrypt 2 rounds in the backward. The words X_0, X_9, X_{18}, X_{28}, X_{39} after a 6-round distinguisher is retrieved. If the guessed 25-bit equivalent key is right, the words X_0, X_9, X_{18}, X_{28}, X_{39} satisfy the integral distinguisher. The filtering probability is $\Pr = 2^{-25}$. We repeat the process 8 times for each diagonal to recover the full 200-bit equivalent key.

Complexity evaluation. In the beginning of the attack, we need 2^{35} chosen plaintext to build the 6-round integral distinguisher. Then, we guess 25-bit equivalent key and check for the validity requiring the time of $25 \cdot 4 \cdot 25 \cdot 2^{25} \simeq 2^{36.29}$ additions. From the filtering probability, two candidates (one right key and one random wrong key) for the guessed equivalent key words are remained on average. After the process, a total of 2^8 candidates are remained for eK after. Since the right key satisfies the relation $0^{40}||K = \mathrm{MC}(\mathrm{eK})$, we check the remaining 2^8 candidates of eK, in which $2^8 \times 2^{-40} < 1$ candidate remained for the key K. Thus, only the right key exists after the attack. Even if we obtain more

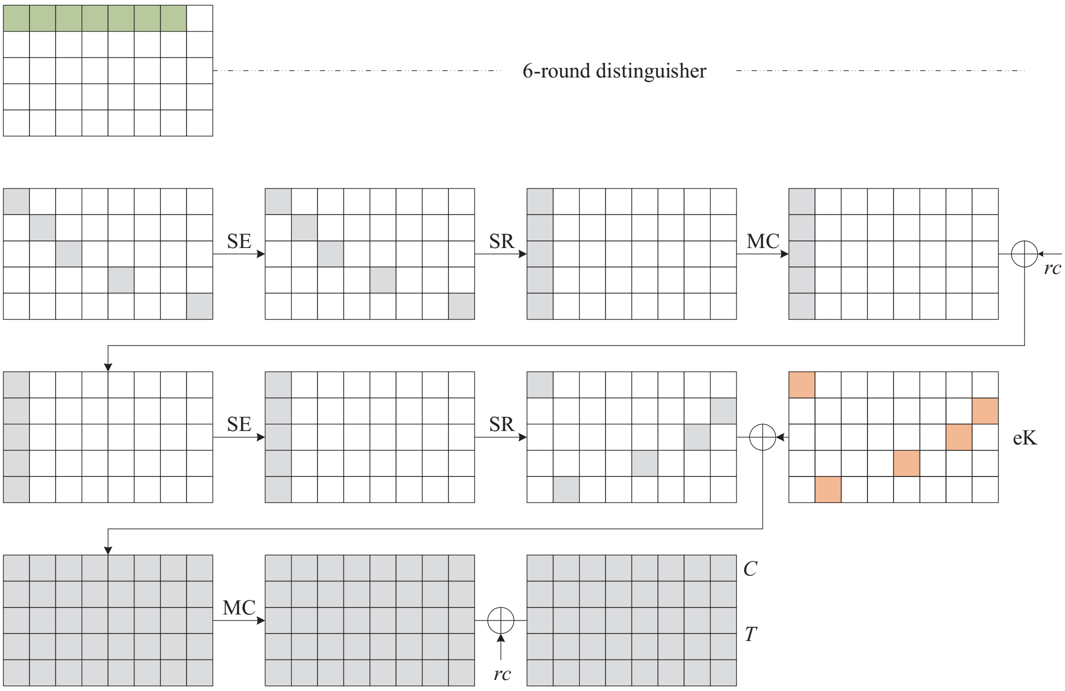

Figure 4 (Color online) The 8-round integral attack on APE-80.

candidates than expected, we could rebuild a different integral distinguisher by choosing another 7 S-boxes to loop through all possible values. This reduces the number of candidates at the expense of slightly increasing the complexity of the attack which is not the main part of the complexity. Hence, the integral attack on 8-round APE-80 requires 2^{35} chosen messages and the time of $8\cdot 25\cdot 4\cdot 25\cdot 2^{25}\simeq 2^{39.29}$ additions.

In addition, if we make state words $X_0,\ldots,X_5$ active and the remain fixed to constant, an improved integral distinguisher in Subsection 3.3 is constructed. With another unaltered part of the attack, we reduce the required data complexity from 2^{35} to 2^{30}.

5.2 Integral attack on APE-120

In the case of APE-120, we obtain an improved 6-round integral distinguisher that requires 2^{30} chosen messages. We use the distinguisher to perform a key-recovery attack on 8-round finalization of APE-120. In the attack process, we need to guess 35 key bits to check the distinguisher, we omit the details here. This attack requires 2^{30} chosen messages and the time of $280\cdot 4\cdot 35\cdot 2^{35}\simeq 2^{50.26}$ additions.

6 Forgery attacks

Based on Propositions 1 and 2, we create forgery for 5/6-round finalization of APE and HANUMAN, requiring the practical complexity $2^{15}/2^{30}$.

For APE, we construct a structure that contains m messages with the same associated data and same number of blocks of plaintexts. Assume that they all have t plaintext blocks, the m messages have the same values for $(t-1)$ plaintext blocks $P_1,\ldots,P_{t-1}$, but differ only in the last block of plaintext P_t. Based on Proposition 1, we make the first 3/6 words of P_t to take over all possible values, showing this structure contains $m=2^{15}/2^{30}$ messages. Let P_j^i, C_j^i denote the j-block of plaintext and ciphertext for i-th message and T^i be the tag of i-th message. For 5/6-round finalization of APE, the following relation holds:

$$C_t^1\oplus C_t^2\oplus\cdots\oplus C_t^m=0,\quad T^1\oplus T^2\oplus\cdots\oplus T^m=0.$$

If the knowledge of ciphertexts last blocks and tags for arbitrary $(m-1)$ messages in the structure are known, we could determine the last block of ciphertext and tag for the remaining message. For example,

$C_t^1, \ldots, C_t^{m-1}$ and $T^1, \ldots, T^{m-1}$ are known; then, $C_t^m = C_t^1 \oplus \cdots \oplus C_t^{m-1}$ and $T^m = T^1 \oplus \cdots \oplus T^{m-1}$.

Similarly, we can create forgeries for 5/6-round finalization version of HANUMAN, that differes from the forgery of APE at the point that the finalization of HANUMAN generates only the tag. We can still create forgeries in a similar and easier manner, requiring complexity $2^{15}/2^{30}$, respectively.

7 Conclusion

We develop the new integral distinguishers for both forward and backward rounds of PRIMATE permutation, estimating the algebraic degree as accurately as possible. Based on these findings, we combine two distinguishers targeting forward rounds and backward rounds, respectively, to build a full 12-round zero-sum distinguisher for PRIMATE-80/120 permutation with the complexity 2^{100} and 2^{105}. Then, a 6-round integral distinguisher is applied to perform key-recovery attacks on APE-80/120 with 2^{30} chosen messages and the time of $2^{39.29}/2^{50.26}$ additions. Our work is the best attack on APE. For the first time, we demonstrate the practical attack against 8-round finalization of APE-80. In addition, the new distinguisher can be used to create forgeries for 5/6-round finalization of APE and HANUMAN with practical complexity $2^{15}/2^{30}$, which are the first forgery attacks on APE and HANUMAN.

Acknowledgements This work was supported by National Cryptography Development Fund (Grant No. MMJJ20170102), National Natural Science Foundation of China (Grant Nos. 61572293, 61502276, 61692276), Major Scientific and Technological Innovation Projects of Shandong Province (Grant No. 2017CXGC0704), National Natural Science Foundation of Shandong Province (Grant No. ZR2016FM22), and Open Research Fund from Shandong Provincial Key Laboratory of Computer Network (Grant No. SDKLCN-2017-04).

References

1 Rogaway P. Authenticated-encryption with associated-data. In: Proceedings of ACM Conference on Computer and Communications Security (CCS), 2002. 98–107
2 Bellare M, Namprempre C. Authenticated encryption: relations among notions and analysis of the generic composition paradigm. In: Proceedings of the 6th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology, 2000. 531–545
3 Jutla C. Encryption modes with almost free message integrity. In: Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques: Advances in Cryptology, 2001. 529–544
4 Gligor V, Donescu P. Fast encryption and authentication: XCBC encryption and XECB authentication modes. In: Proceedings of International Workshop on Fast Software Encryption, 2002. 92–108
5 Rogaway P, Bellare M, Black J, et al. OCB: a block-cipher mode of operation for efficient authenticated encryption. In: Proceedings of ACM Transactions on Information and System Security (TISSEC), 2003. 365–403
6 National Institute of Standards and Technology (NIST). Advanced encryption standard (AES). FIPS 197. https://csrc.nist.gov/csrc/media/publications/fips/197/final/documents/fips-197.pdf
7 Dworkin M. Recommendation for block cipher modes of operation: Galois/counter mode (GCM) and GMAC. NIST Special Publication 800-38D, 2007. https://www.govinfo.gov/content/pkg/GOVPUB-C13-1e1d0b2a761f50d919d892b9e020965b/pdf/GOVPUB-C13-1e1d0b2a761f50d919d892b9e020965b.pdf
8 The CAESAR Committee. CAESAR: competition for authenticated encryption: security, applicability, and robustness. http://competitions.cr.yp.to/caesar.html
9 Andreeva E, Bilgin B, Bogdanov A, et al. PRIMATEs v1.02: submission to the CAESAR competition. http://primates.ae/
10 Saha D, Kuila S, Chowdhury D R. EscApe: diagonal fault analysis of APE. In: Proceedings of International Conference on Cryptology in India, 2014. 197–216
11 Minaud B. Improved beer-recovery attack against APE. https://aezoo.compute.dtu.dk/doku.php?id=primates
12 Morawiecki P, Pieprzyk J, Srebrny M, et al. Applications of key recovery cube-attack-like. 2015. http://eprint.iacr.org/2015/1009.pdf
13 Lukas K, Daemen J. Cube attack on primates. Proc Rom Acad, 2017, 18: 293–306
14 Todo Y, Aoki K. FFT key recovery for integral attack. In: Proceedings of International Conference on Cryptology and Network Security, 2014. 64–81
15 Daemen J, Rijmen V. The wide trail design strategy. In: Proceedings of the 8th IMA International Conference on Cryptography and Coding, 2001. 222–238
16 Daemen J, Rijmen V. The Design of Rijndael. Berlin: Springer, 2002
17 Boura C, Canteaut A. A zero-sum proposition for the Keccak-f permutation with 18 rounds. In: Proceedings of IEEE International Symposium on Information Theory, 2010. 2488–2492
18 Yang M H, Lai X J. The computational method of the algebraic degree of Boolean functions (in Chinese). In: Proceedings of Annual Meeting of Chinese Association for Cryptologic Research, 2009
19 Aumasson J P, Meier W. Zero-sum distinguishers for reduced Keccak-f and for the core functions of Luffa and Hamsi. 2009. http://www.131002.net/data/papers/AM09.pdf