Postprint: Search for Integral Distinguishers of PRIDE and RoadRunneR Based on Bit-based Division Property

Authors: Li Yanjun, Zhao Jingming

Date: 2018-11-29T00:00:00+00:00

Abstract

PRIDE and RoadRunneR are two lightweight block cipher algorithms proposed in recent years. At Asiacrypt 2016, Xiang Zejun et al. proposed utilizing MILP (Mixed Integer Linear Programming) models based on bit-based division property to search for integral distinguishers. Building upon this idea, to evaluate the integral properties of these algorithms and verify the practicality of the new method, MILP models are constructed separately according to their different cryptographic structures and solved using the Gurobi optimizer to search for usable integral distinguishers. The results yield 9-round and 5-round integral distinguishers, respectively, which represent the longest known integral distinguishers for PRIDE and RoadRunneR to date, enabling integral attacks on more rounds.

Full Text

Preamble

Vol. 37 No. 1
Application Research of Computers
ChinaXiv Cooperative Journal

Integral Distinguisher Search of PRIDE and RoadRunneR Based on Bit-Based Division Property

Li Yanjun, Zhao Jingming
(Department of Information Security, Beijing Electronic Science and Technology Institute, Beijing 100070, China)

Abstract: PRIDE and RoadRunneR are two lightweight block cipher algorithms proposed in recent years. At ASIACRYPT 2016, Xiang Zejun et al. proposed using MILP (Mixed Integer Linear Programming) models based on bit-

based division property to search for integral distinguishers. Applying this approach to two different types of lightweight block cipher algorithms, we construct MILP models according to their distinct structures to evaluate their integral properties and verify the practicality of this new method. By solving these models using the Gurobi optimizer, we successfully search for usable integral distinguishers. The results yield 9-round and 5-round integral distinguishers for PRIDE and RoadRunneR, respectively, which represent the longest known integral distinguishers for these ciphers and enable integral attacks on more rounds.

Keywords: PRIDE; RoadRunneR; bit-based division; MILP model; integral distinguisher

---

1.1 Introduction to PRIDE and RoadRunneR Algorithms

PRIDE is a lightweight block cipher with a 64-bit block length and 128-bit key length, proposed by Albrecht et al. at CRYPTO 2014. It employs an SPN structure with 20 rounds. The 64-bit round function input is divided into 16 nibbles, which are XORed with round keys, then processed in parallel through the S-layer consisting of 16 identical S-boxes (see Table 1), followed by a linear layer. The round function structure is illustrated in Figure 1 [Figure 1: see original paper]. The specific matrix representation and additional details of PRIDE can be found in reference [1].

RoadRunneR, proposed in 2015, is a lightweight block cipher that adopts a Feistel structure with an SPN-type round function. It has a 64-bit block length and supports key lengths of 80/128 bits, iterating for 10/12 rounds respectively. The round function F consists of four SLK functions plus one S-layer, with the S-box shown in Table 2 . The S, L, and K layers represent the substitution layer, linear layer, and round key addition layer, respectively. The algorithm structure and round function are depicted in Figure 2 [Figure 2: see original paper], where the left side shows the overall Feistel structure, the upper-right shows the SPN-type round function F, and the lower-right shows the SLK function. Further details about RoadRunneR can be found in reference [2].

2.1 Notation

Let $\mathbb{F}_2$ denote the binary finite field and $\mathbb{F}_2^n$ denote the set of n-bit vectors over $\mathbb{F}_2$. Let $\mathbb{Z}$ and $\mathbb{Z}^n$ denote the integer ring and the set of n-dimensional integer vectors, respectively. For any $a \in \mathbb{F}_2^n$, $a[i]$ represents the i-th element of a, and $w(a)$ denotes the Hamming weight calculated as $w(a) = \sum_{i=0}^{n-1} a[i]$. For any vector $a = (a_0, a_1, \dots, a_{m-1}) \in \mathbb{F}_2^{n_0} \times \mathbb{F}_2^{n_1} \times \cdots \times \mathbb{F}_2^{n_{m-1}}$, the Hamming weight is defined as $w(a) = \sum_{i=0}^{m-1} w(a_i)$. Let $k = (k_0, k_1, \dots, k_{m-1})$ and $k^* = (k_0^*, k_1^*, \dots, k_{m-1}^*)$ be

two vectors in $\mathbb{Z}^m$. We define $k \succeq k^*$ if and only if all corresponding components satisfy $k_i \geq k_i^*$ for $i = 0, 1, \ldots, m-1$.

The bit product function $\pi_u(x)$ is a function from $\mathbb{F}_2^n$ to $\mathbb{F}_2$ defined for any $u \in \mathbb{F}_2^n$ and $x \in \mathbb{F}_2^n$ as:

$$\pi_u(x) = \prod_{i=0}^{n-1} x[i]^{u[i]} = \prod_{u[i]=1} x[i]$$

For any vector $u = (u_0, u_1, \ldots, u_{m-1}) \in \mathbb{F}_2^{n_0} \times \mathbb{F}_2^{n_1} \times \cdots \times \mathbb{F}_2^{n_{m-1}}$, we define:

$$\pi_u(x) = \prod_{i=0}^{m-1} \pi_{u_i}(x_i)$$

2.2 Division Property and Division Trails

Definition 1 (Division Property). Let X be a multiset whose elements take values from $\mathbb{F}_2^n$. When X satisfies the division property $\mathcal{D}_k^{n,m}$, where k is an m-dimensional vector with each component taking integer values from 0 to n, the following condition holds: for any $u \in \mathbb{F}_2^{n_0} \times \mathbb{F}_2^{n_1} \times \cdots \times \mathbb{F}_2^{n_{m-1}}$, the parity $\bigoplus_{x \in X} \pi_u(x)$ is always even when $u \in W_k = \{u \in \mathbb{F}_2^{n_0} \times \cdots \times \mathbb{F}_2^{n_{m-1}} \mid w(u) \not\succeq k\}$.

Division property was introduced by Todo at EUROCRYPT 2015 as a generalization of integral properties [3]. At FSE 2016, Todo and Morii proposed bit-based division property [5] and applied it to find a 14-round integral distinguisher for SIMON32. At ASIACRYPT 2016, Xiang Zejun et al. proposed a method for searching integral distinguishers using MILP models based on bit-based division property [6], overcoming the enormous time and space overhead of direct bit-based division property searches. By selecting appropriate objective functions and accurately modeling division property propagation, they analyzed six lightweight block ciphers with bit-permutation diffusion layers, leaving the search for algorithms with non-bit-permutation diffusion layers as an open problem. Sun Ling et al. addressed this issue by introducing intermediate variables and establishing $2n$ linear inequalities to represent division trails through an n-bit input linear layer [7].

Definition 2 (Division Trail). Let f be the round function of a block cipher. Assuming the initial input division property is $\mathcal{D}_{k_0}^{n,m}$, and after r rounds of propagation the output division property is $\mathcal{D}_{k_r}^{n,m}$, we obtain the following division property propagation chain:

$$\mathcal{D}_{k_0}^{n,m} \xrightarrow{f} \mathcal{D}_{k_1}^{n,m} \xrightarrow{f} \cdots \xrightarrow{f} \mathcal{D}_{k_r}^{n,m}$$

If there exists a vector k_i in the chain that does not belong to $\mathcal{D}_{k_{i+1}}^{n,m}$, then there must exist a vector k_i^* in $\mathcal{D}_{k_i}^{n,m}$ that corresponds to it. We can say that k_i^*

can propagate to k_{i+1} through division property propagation rules, denoted as $k_i \xrightarrow{f} k_{i+1}$. This paper refers to such a chain as an r-round division trail.

The general steps for searching integral distinguishers using MILP models based on bit-based division property are: first, specify the initial bit-based division property by designating active and inactive bits; second, establish a model representing division trails through the round function, including through substitution and diffusion layers. Currently, the representation method for division trails through S-boxes is well-established, while representation methods for linear layers remain under investigation [9]. Finally, select an appropriate objective function as the search termination condition to determine whether a usable integral distinguisher exists.

2.3 Bit-Based Division Property Propagation Rules and Modeling

Todo proved the propagation rules for traditional division property [3] and summarized them in [4] as five operations: substitution, copy, XOR, split, and merge. In bit-based division property, only the copy and XOR propagation rules are used. For S-boxes (substitution operations), more detailed study is required, which was addressed in [8][10]. Algorithm 2 in [6] provides a general algorithm for computing bit-based division property propagation through S-boxes.

Rule 1 (Copy). Let F be a copy function with input $x \in \mathbb{F}_2$ and outputs represented as $(y_0, y_1) = (x, x)$. Let X and Y be the corresponding input and output sets. If X satisfies division property $\mathcal{D}_{k_0}$, then Y satisfies division property $\mathcal{D}_{k_1}$. The propagation process has only two possible cases.

Rule 2 (XOR Compression). Let F be an XOR compression function with inputs $(x_0, x_1) \in \mathbb{F}_2^2$ and output represented as $y = x_0 \oplus x_1$. Let X and Y be the corresponding input and output sets. If X satisfies division property $\mathcal{D}_{k_0}$, then Y satisfies division property $\mathcal{D}_{k_1}$. The propagation process has only four possible cases.

Rule 3 (AND Compression). Let F be an AND compression function with inputs $(x_0, x_1) \in \mathbb{F}_2^2$ and output represented as $y = x_0 \wedge x_1$. Let X and Y be the corresponding input and output sets. If X satisfies division property $\mathcal{D}_{k_0}$, then Y satisfies division property $\mathcal{D}_{k_1}$. The propagation process has only four possible cases.

The following briefly explains the modeling process for division property propagation through copy, AND, and XOR operations using linear inequality systems.

Model 1 (Copy). Let $(a) \xrightarrow{\text{copy}} (b_0, b_1)$ be a division trail of the copy function. The following inequality system accurately represents the division property

propagation of the copy operation:

$$\begin{cases} b_0 + b_1 - a = 0 \\ a, b_0, b_1 \text{ are binary numbers} \end{cases}$$

Model 2 (AND). Let $(a_0, a_1) \xrightarrow{\text{AND}} (b)$ be a division trail of the bit-AND function. The following inequality system accurately represents the division property propagation of the AND operation:

$$\begin{cases} b - a_0 \geq 0 \\ b - a_1 \geq 0 \\ a_0 + a_1 - b \geq 0 \\ a_0, a_1, b \text{ are binary numbers} \end{cases}$$

Model 3 (XOR). Let $(a_0, a_1) \xrightarrow{\text{XOR}} (b)$ be a division trail of the bit-XOR function. The following inequality system accurately represents the division property propagation of the XOR operation:

$$\begin{cases} a_0 + a_1 - b = 0 \\ a_0, a_1, b \text{ are binary numbers} \end{cases}$$

Modeling S-boxes. For S-box propagation, Xiang Zejun et al. provided a general algorithm (Algorithm 2 in [6]) to compute the bit-based division property propagation through S-boxes. The Sage software's `Inequality_{generator}()` function is then used to obtain a set of linear inequalities representing these propagation rules, which can be further reduced using inequality reduction algorithms [6].

2.4 Objective Function

If a set X satisfies division property $\mathcal{D}_k^{n,m}$, it has no integral property if and only if k contains all n unit vectors. Let $k^{(i)}$ denote the output division property vector after i rounds of encryption. The propagation terminates when $k^{(i)}$ first contains all n unit vectors, yielding an r-round distinguisher. To determine whether an $(r-1)$-round distinguisher exists, we only need to check whether $k^{(r-1)}$ contains all unit vectors. Therefore, the objective function is set to minimize the sum of components of the last vector $k^{(r)}$ in any r-round division trail:

$$\text{Obj: Minimize} \sum_{i=0}^{n-1} k_i^{(r)}$$

---

3 PRIDE' s MILP Model

In division property propagation, XOR with constants does not change the division property, meaning round key addition does not affect propagation. Therefore, we only consider the effects of S-boxes and linear diffusion layers on division property propagation.

3.1 Linear Inequality Representation of the S-box Layer

PRIDE' s S-layer consists of 16 identical S-boxes in parallel. We first study division trails through a single S-box. Let the S-box input be (x_3, x_2, x_1, x_0) and output be (y_3, y_2, y_1, y_0). The Algebraic Normal Form (ANF) representation of the S-box is:

$$\begin{cases} y_0 = x_0 \oplus x_1 \oplus x_2 \oplus x_3 \\ y_1 = x_0 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_{0x}1 \oplus x_{0x}2 \oplus x_{0x}3 \oplus x_{1x}2 \oplus x_{1x}3 \\ y_2 = x_0 \oplus x_2 \oplus x_{0x}1 \oplus x_{0x}2 \oplus x_{0x}3 \oplus x_{1x}2 \oplus x_{1x}3 \\ y_3 = x_0 \oplus x_1 \oplus x_3 \oplus x_{0x}1 \oplus x_{0x}2 \oplus x_{0x}3 \oplus x_{1x}2 \oplus x_{1x}3 \end{cases}$$

Applying Algorithm 2 yields 44 division trails through the S-box, shown in Table 3 . Using Sage's `Inequality_{generator}()` function, these 44 trails are represented by 110 linear inequalities, which are then reduced to 11 inequalities. Let $(a_0, a_1, a_2, a_3) \xrightarrow{S} (b_0, b_1, b_2, b_3)$ denote a division trail through an S-box. The linear inequality representation is:

For the entire S-layer, we concatenate sixteen 4-dimensional input vectors, with outputs similarly represented as sixteen 4-dimensional vectors. Let $(a_0, a_1, \dots, a_{63}) \xrightarrow{S} (b_0, b_1, \dots, b_{63})$ denote a division trail through the S-layer. If the S-layer input set satisfies division property $\mathcal{D}_{k_0}^{64,16}$, then the output satisfies division property $\mathcal{D}_{k_1}^{64,16}$.

Additionally, the copied bits from the S-box output need to be XORed with relevant output bits. Since variables in the same row need to be XORed with identical variables, to describe all XOR operations, we generate 16 linear inequalities:

Given an S-layer input set satisfying division property $\mathcal{D}_{k_0}^{64,16}$, the output after the S-layer satisfies division property $\mathcal{D}_{k_1}^{64,16}$.

3.2 Linear Inequality Representation of the Linear Layer

Using MILP models to search for ciphers with more complex linear layers was an open problem. This can be addressed by introducing intermediate variables: any linear matrix layer can be decomposed into copy and XOR operations [7].

PRIDE's 64×64 matrix layer consists of four 16×16 sub-matrices arranged diagonally: L_0, L_1, L_2, L_3. Taking L_0 as an example, we derive the linear inequality

system for L_0, with similar processes for the other three matrices.

Assume the input set passing through matrix L_0 satisfies division property $\mathcal{D}_{k_0}^{16,4}$, where $k_0 = (x_0, x_1, \dots, x_{15})$. From the first column of L_0, we see that x_0 is copied 5 times. These copied values are then XORed with other copied values in different rows. The remaining operations for x_1, x_2, x_3 are similar.

L_0 contains 48 non-zero elements. By introducing intermediate variables $t_0, t_1, \dots, t_{47}$, we transform it into the following form:

To describe all copy operations in L_0, we generate 16 linear inequalities:

To obtain division trails through L_0, we simply combine the two linear inequality systems (1) and (2) into one system.

Similarly, division trails through L_1, L_2, L_3 are represented as $(x_{16}, \dots, x_{31}) \xrightarrow{L_1} (y_{16}, \dots, y_{31})$, $(x_{32}, \dots, x_{47}) \xrightarrow{L_2} (y_{32}, \dots, y_{47})$, and $(x_{48}, \dots, x_{63}) \xrightarrow{L_3} (y_{48}, \dots, y_{63})$, with intermediate variables $t_{48}, \dots, t_{95}$, $t_{96}, \dots, t_{143}$, and $t_{144}, \dots, t_{191}$, respectively. Following the propagation rules above, we generate corresponding linear inequality systems.

Let $(x_0, x_1, \dots, x_{63}) \xrightarrow{L} (y_0, y_1, \dots, y_{63})$ denote a division trail through the linear matrix layer L. This inequality system is reduced to 10 inequalities. Given a linear layer input set satisfying division property $\mathcal{D}_{k_0}^{64,16}$, the output after the linear layer satisfies division property $\mathcal{D}_{k_1}^{64,16}$.

Having obtained the linear inequality representations for division trails through both the S-layer and L-layer, we combine them to get division trails for the round function. Repeating the round function r times yields an r-round division trail.

---

4 RoadRunneR' s MILP Model

RoadRunneR adopts a Feistel structure overall, containing copy and XOR operations that can be modeled using the methods described in Section 2.3. While the round function is complex, it fundamentally consists of S-layer and linear diffusion layers, allowing modeling similar to the steps in Section 3.

RoadRunneR uses whitening keys in the first and last rounds and incorporates round constant XOR operations in each round. However, since XOR with constants does not change division property, these operations are ignored in our model.

4.1 Linear Inequality Representation of the S-box Layer

RoadRunneR' s S-layer consists of 8 identical S-boxes in parallel. Let the S-box input be (x_3, x_2, x_1, x_0) and output be (y_3, y_2, y_1, y_0). The ANF representation

of the S-box is:

$$\begin{cases} y_0 = x_0 \oplus x_1 \oplus x_2 \oplus x_3 \\ y_1 = x_0 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_{0x}1 \oplus x_{0x}2 \oplus x_{0x}3 \oplus x_{1x}2 \oplus x_{1x}3 \\ y_2 = x_0 \oplus x_2 \oplus x_{0x}1 \oplus x_{0x}2 \oplus x_{0x}3 \oplus x_{1x}2 \oplus x_{1x}3 \\ y_3 = x_0 \oplus x_1 \oplus x_3 \oplus x_{0x}1 \oplus x_{0x}2 \oplus x_{0x}3 \oplus x_{1x}2 \oplus x_{1x}3 \end{cases}$$

Applying Algorithm 2 yields 43 division trails through the S-box, shown in Table 4 . Using Sage' s `Inequality_{generator}()` function, these 43 trails are represented by 132 linear inequalities, which are reduced using inequality reduction algorithms.

4.2 Linear Inequality Representation of the Linear Layer

RoadRunneR's matrix layer consists of four identical 8×8 matrices L in parallel. We derive the linear inequality system for L as follows.

Assume the input set passing through matrix L satisfies division property $\mathcal{D}_{k_0}^{8,4}$, where $k_0 = (x_0, x_1, \ldots, x_7)$. Matrix L contains 24 non-zero elements. By introducing intermediate variables $t_0, t_1, \ldots, t_{23}$, we transform it into the following form:

To describe all copy operations in L, we generate 8 linear inequalities:

To describe all XOR operations in L, we generate 8 linear inequalities:

Let $(x_0, x_1, \ldots, x_7) \xrightarrow{L} (y_0, y_1, \ldots, y_7)$ denote a division trail through L. If a linear layer input set satisfies division property $\mathcal{D}_{k_0}^{32,4}$, then the output after the linear layer satisfies division property $\mathcal{D}_{k_1}^{32,4}$.

Having obtained the linear inequality representations for division trails through both the S-layer and L-layer, we combine them to get division trails for the SLK function. Combining three SLK functions and one S-layer yields a division trail for one round. Repeating the round function r times gives an r-round division trail.

---

5 Integral Distinguisher Search

With the complete MILP models for PRIDE and RoadRunneR established, we use the constraints and objective function described above. Following the search algorithm from [6] and applying the Gurobi optimizer to solve these models, we can search for the existence of r-round integral distinguishers. All experimental data in this section were obtained on the following platform: Intel(R) Celeron(R) CPU 1007U @ 1.50 GHz, 4 GB RAM, 64-bit Windows 7. The search was implemented in Python, yielding the results shown in Table 5 .

Table 5. Searched Integral Distinguishers

Cipher	Block Length	Distinguisher Rounds	Active Bits	Balanced Bits
PRIDE	64 bits	9 rounds	60 bits	32 bits
RoadRunneR	64 bits	5 rounds	50 bits	32 bits

5.1 PRIDE' s 8-Round Integral Distinguisher

If we select the initial division property as $\mathcal{D}_{k_0}^{64,16}$ where the rightmost 60 bits of the input are active and the remaining 4 bits are constant, we obtain a valid 8-round distinguisher where the rightmost 32 bits of the output are balanced. The 8-round distinguisher for PRIDE can be represented as: [description of distinguisher pattern].

5.2 PRIDE' s 9-Round Integral Distinguisher

If we select the initial division property as $\mathcal{D}_{k_0}^{64,16}$ where the rightmost 63 bits are active and 1 bit is constant, we obtain a valid 9-round distinguisher where the rightmost bit of the output is balanced. The 9-round distinguisher for PRIDE can be represented as: [description of distinguisher pattern].

5.3 RoadRunneR' s 5-Round Integral Distinguisher

If we select the initial division property as $\mathcal{D}_{k_0}^{64,2}$ where all 32 bits of the right half are active, 18 bits of the left half are active, and the remaining 14 bits are constant, we obtain a valid 5-round distinguisher where the rightmost 32 bits of the output are balanced. The 5-round distinguisher for RoadRunneR can be represented as: [description of distinguisher pattern].

We also found a 4-round distinguisher for RoadRunneR, which matches the 4-round distinguisher proposed by the algorithm designers in the original paper, confirming the correctness of our search method.

---

6 Conclusion

Previous work has analyzed PRIDE using differential and linear cryptanalysis [13][14], but no integral analysis of PRIDE had been conducted. RoadRunneR' s designers proposed a 4-round distinguisher in the original paper. This paper uses MILP models based on bit-based division property to search for integral distinguishers of PRIDE, obtaining a 9-round distinguisher—the longest known integral distinguisher for PRIDE. Applying this method to RoadRunneR yields a 5-round distinguisher, improving upon the designers' result by one round. This demonstrates that for both Feistel and SPN-type block ciphers, this technique can find more effective integral distinguishers compared to traditional methods.

Using these longest-known integral distinguishers enables the most round-efficient integral attacks on these ciphers. For example, with PRIDE' s 8-round distinguisher, we can extend two rounds forward to attack 10 rounds and recover the corresponding key using related-key recovery techniques. However, increasing the distinguisher length also increases the required number of chosen plaintexts, which does not significantly reduce the data complexity of the overall attack. For instance, the 9-round distinguisher for PRIDE requires 2^{60} chosen plaintexts. Therefore, reducing the number of chosen plaintexts while maintaining the same distinguisher length remains a challenge. Current research suggests that integral distinguishers found using bit-based division property are often optimal, though not necessarily guaranteed to be optimal, depending on the accuracy of the propagation modeling method—this represents a primary direction for future research.

As research on bit-based division property and automated distinguisher searching progresses, combining mathematical models with optimization software has significantly improved search efficiency and advanced integral cryptanalysis. The potential of bit-based division property for finding distinguishers remains substantial, and this paper contributes to the study and application of this method.

References

[1] Albrecht M, Driessen B, Kavun E B, et al. Block ciphers focus on the linear layer (feat. PRIDE) [C/OL]. IACR Cryptology ePrint Archive, (2014). https://eprint.iacr.org/2014/453.pdf.

[2] Baysal A, Sahin S. RoadRunneR: A small and fast bitslice block cipher for low cost 8-bit processors [C/OL]. LightSec Cryptology ePrint Archive. (2015). https://eprint.iacr.org/2015/906.pdf.

[3] Todo Y. Structural evaluation by generalized integral property [C]// Proc of EUROCRYPT, LNCS, vol. 9056. Berlin: Springer, 2015: 287–314.

[4] Todo Y. Integral cryptanalysis on full MISTY1 [C]// Proc of CRYPTO Part I. LNCS vol. 9215. Berlin: Springer, 2015: 413–432.

[5] Todo Y, Morii M. Bit-based division property and application to SIMON family [C]// Pre-Proceedings of FSE, 2016. Berlin: Springer, 2015: 357–377.

[6] Xiang Zejun, Zhang Wentao, Bao Zhenzhen et al. Applying MILP method to search integral distinguishers based on division property for 6 lightweight block ciphers [C]// Advances in Cryptology. Berlin: Springer, 2016: 648-678.

[7] Sun Ling, Wang Wei, Wang Meiqin. MILP-aided bit-based division property for primitives with non-bit-permutation linear layers [C/OL]. Cryptology ePrint Archive. (2016) http://eprint.iacr.org/2016/811.

[8] Sun Ling, Wang Meiqin. Towards a further understanding of bit-based division property [J]. Science China Information Sciences, 2017, 60 (12).

[9] Zhang Wenying, Rijmen V. Division cryptanalysis of block ciphers with a binary diffusion layer [C/OL]. IACR Cryptology ePrint Archive. (2017) https://eprint.iacr.org/2017/188.

[10] Boura C, Canteaut A. Another view of the division property [C]// Advances in Cryptology-CRYPTO. Berlin: Springer, 2016: 654-680.

[11] Daemen J, Knudsen L, Rijmen V. The block cipher square [C]// Proc of the 4th International Workshop on Fast Software Encryption. Berlin: Springer, 1997: 149-165.

[12] Knudsen L, Wagner D. Integral cryptanalysis [C]// Proc of the 9th Int Workshop on Fast Software Encryption. Berlin: Springer, 2002.

[13] Dai Yibin, Chen Shaozhen. Cryptanalysis of full PRIDE block cipher [J]. Science China Information Sciences. 2017, 60 (5): 052108.

[14] Yi Wentan, Chen Shaozhen, Tian Ya. Linear cryptanalysis of reduced-round PRIDE block cipher [J]. Acta Electronica Sinica, 2017, 45 (2): 468-476.

[15] Yu Xiaoli, Wu Wenling, Li Yanjun. Integral attack of reduced-round MIBS block cipher [J]. Journal of Computer Research & Development, 2013, 50 (10): 2117-2125.

Note: Figure translations are in progress. See original paper for figures.