Wu WL, Zhang WT, Feng DG. Impossible differential cryptanalysis of reduced-round ARIA and Camellia. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 22(3): 449~456 May 2007

Impossible Differential Cryptanalysis of Reduced-Round ARIA and Camellia

Wen-Ling Wu[1] (吴文玲), Wen-Tao Zhang[2] (张文涛), and Deng-Guo Feng[1] (冯登国)

[1]*State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences, Beijing 100080, China*

[2]*State Key Laboratory of Information Security, Graduate University of Chinese Academy of Sciences, Beijing 100080, China*

E-mail: {wwl, feng}@is.iscas.ac.cn; zhangwt@gucas.ac.cn

Received October 1, 2006; revised March 12, 2007.

Abstract This paper studies the security of the block ciphers ARIA and Camellia against impossible differential cryptanalysis. Our work improves the best impossible differential cryptanalysis of ARIA and Camellia known so far. The designers of ARIA expected no impossible differentials exist for 4-round ARIA. However, we found some nontrivial 4-round impossible differentials, which may lead to a possible attack on 6-round ARIA. Moreover, we found some nontrivial 8-round impossible differentials for Camellia, whereas only 7-round impossible differentials were previously known. By using the 8-round impossible differentials, we presented an attack on 12-round Camellia without FL/FL^{-1} layers.

Keywords block cipher, ARIA, Camellia, data complexity, time complexity, impossible differential cryptanalysis

1 Introduction

Both ARIA[1] and Camellia[2] support 128-bit block size and 128-, 192-, and 256-bit key lengths, i.e., the same interface specifications as the Advanced Encryption Standard (AES). Camellia was jointly developed in 2000 by Nippon Telegraph and Telephone Corporation (NTT) and Mitsubishi Electric Corporation (Mitsubishi). It has now been selected as an international standard by ISO/IEC, and also been adopted by cryptographic evaluation projects such as NESSIE and CRYPTREC, as well as the standardization activities at IETF. It means Camellia gradually become one of the most worldwide used block ciphers. Therefore, a constant evaluation of its security with respect to various cryptanalytic techniques is required. Camellia was already analyzed in many papers using various attacks[3~10].

ARIA was designed by a group of Korean experts in 2003. In 2004, ARIA was established as a Korean Standard block cipher algorithm (KS X 1213) by the Ministry of Commerce, Industry and Energy. ARIA is a general-purpose involutional SPN block cipher algorithm, optimized for lightweight environments and hardware implementation. Its security was analyzed initially by the designers internally, and later by the COSIC group of K.U. Leuven, Belgium[11]. They analyzed the security of ARIA against differential and linear cryptanalysis[12,13], truncated and higher-order differential[14], impossible differential[15], slide attack[16,17], integral attack[18], and other attacks[19~21].

Impossible differential means a differential that holds with probability 0, or a differential that does not exist. Impossible differential attacks use impossible differentials to derive the actual values of the keys, which has been used to attack AES and get very good results[22~27].

In this paper, we examine the security of ARIA and Camellia against impossible differential attacks. The initial analysis of the security of Camellia to impossible differential Cryptanalysis was given in [4]. They presented some nontrivial 7-round impossible differentials for Camellia. We found some nontrivial 8-round impossible differentials, which may lead to a possible attack of Camellia reduced to 12 rounds without FL/FL^{-1}, the attack having complexity less than that of exhaustive search to 12-round Camellia without FL/FL^{-1} layers.

As for ARIA, the designers expected that there was no impossible differentials on 4 or more rounds in [1, 28]. In this paper, we found some 4-round impossible differentials, which lead to a possible attack of ARIA reduced to 6 rounds. The attack requires 2^{121} plaintext/ciphertext pairs and 2^{112} encryptions.

The contents of this paper are as follows. In Section 2 we give a brief description of ARIA and Camellia. In Section 3 we describe some 4-round ARIA impossible differentials and the impossible differential attack on 6-round ARIA. In Section 4, we describe some 8-round Camellia impossible differentials and present the impossible differential attack on 12-round Camellia without FL/FL^{-1} layers. Finally, Section 5 summarizes this paper.

2 ARIA and Camellia

Due to space limitation, we only introduce ARIA and Camellia briefly. Details are shown in [1, 2].

2.1 Description of ARIA

ARIA is a substitution permutation network (SPN) and uses an involutional binary 16×16 matrix in its diffusion layer. The 128-bit plaintexts are treated as byte matrices of size 4×4 as the following:

Regular Paper

This work is supported by the National Natural Science Foundation of China under Grant No.90604036; the National Grand Fundamental Research 973 Program of China under Grant No.2004CB318004.

0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

Every round applies three operations to the state matrix:

Round Key Addition (*RKA*). This is done by XORing the 128-bit round key.

Substitution Layer (*SL*). Applying the 8×8 S-boxes 16 times in parallel on each byte. There are two types of substitution layers to be used so as to make the cipher involution.

Diffusion Layer (*DL*). A linear map $A : (F_2^8)^{16} \rightarrow (F_2^8)^{16}$ is given by

$$(x_0|x_1|\ldots|x_{15}) \mapsto (y_0|y_1|\cdots|y_{15}),$$

where

$$\begin{aligned}
y_0 &= x_3 \oplus x_4 \oplus x_6 \oplus x_8 \oplus x_9 \oplus x_{13} \oplus x_{14},\\
y_1 &= x_2 \oplus x_5 \oplus x_7 \oplus x_8 \oplus x_9 \oplus x_{12} \oplus x_{15},\\
y_2 &= x_1 \oplus x_4 \oplus x_6 \oplus x_{10} \oplus x_{11} \oplus x_{12} \oplus x_{15},\\
y_3 &= x_0 \oplus x_5 \oplus x_7 \oplus x_{10} \oplus x_{11} \oplus x_{13} \oplus x_{14},\\
y_4 &= x_0 \oplus x_2 \oplus x_5 \oplus x_8 \oplus x_{11} \oplus x_{14} \oplus x_{15},\\
y_5 &= x_1 \oplus x_3 \oplus x_4 \oplus x_9 \oplus x_{10} \oplus x_{14} \oplus x_{15},\\
y_6 &= x_0 \oplus x_2 \oplus x_7 \oplus x_9 \oplus x_{10} \oplus x_{12} \oplus x_{13},\\
y_7 &= x_1 \oplus x_3 \oplus x_6 \oplus x_8 \oplus x_{11} \oplus x_{12} \oplus x_{13},\\
y_8 &= x_0 \oplus x_1 \oplus x_4 \oplus x_7 \oplus x_{10} \oplus x_{13} \oplus x_{15},\\
y_9 &= x_0 \oplus x_1 \oplus x_5 \oplus x_6 \oplus x_{11} \oplus x_{12} \oplus x_{14},\\
y_{10} &= x_2 \oplus x_3 \oplus x_5 \oplus x_6 \oplus x_8 \oplus x_{13} \oplus x_{15},\\
y_{11} &= x_2 \oplus x_3 \oplus x_4 \oplus x_7 \oplus x_9 \oplus x_{12} \oplus x_{14},\\
y_{12} &= x_1 \oplus x_2 \oplus x_6 \oplus x_7 \oplus x_9 \oplus x_{11} \oplus x_{12},\\
y_{13} &= x_0 \oplus x_3 \oplus x_6 \oplus x_7 \oplus x_8 \oplus x_{10} \oplus x_{13},\\
y_{14} &= x_0 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_9 \oplus x_{11} \oplus x_{14},\\
y_{15} &= x_1 \oplus x_2 \oplus x_4 \oplus x_5 \oplus x_8 \oplus x_{10} \oplus x_{15}.
\end{aligned}$$

Note that the Diffusion layer of the last round is replaced by a round key addition. We shall assume that the 6-round ARIA also has the diffusion layer replaced by a round key addition at the last round.

2.2 Description of Camellia

Camellia is based on the Feistel structure and has 18 rounds (for 128-bit keys) or 24 rounds (for 192/256-bit keys). The FL/FL^{-1} function layer is inserted at every 6 rounds. Before the first round and after the last round, there are pre- and post-whitening layers which use bitwise exclusive-or operations with 128-bit round subkeys, respectively. In this paper, we will consider Camellia without FL/FL^{-1} function layer and whitening layers.

Let L_{r-1} and R_{r-1} be the left and the right halves of the r-th round input, and k_r be the r-th round subkey. Then the Feistel structure of Camellia can be written as

$$L_r = R_{r-1} \oplus F(L_{r-1}, k_r), \quad R_r = L_{r-1},$$

where F is the round function defined below:

$$\begin{aligned}
F : \{0,1\}^{64} \times \{0,1\}^{64} &\longrightarrow \{0,1\}^{64}\\
(X, k_r) &\mapsto Z = P(S(X \oplus k_r)),
\end{aligned}$$

where S and P are defined as follows:

$$\begin{aligned}
&S : (F_2^8)^8 \longrightarrow (F_2^8)^8\\
&x_1|x_2|x_3|x_4|x_5|x_6|x_7|x_8 \mapsto y_1|y_2|y_3|y_4|y_5|y_6|y_7|y_8\\
&y_1 = s_1(x_1), \quad y_2 = s_2(x_2), \quad y_3 = s_3(x_3),\\
&y_4 = s_4(x_4), \quad y_5 = s_2(x_5), \quad y_6 = s_3(x_6),\\
&y_7 = s_4(x_7), \quad y_8 = s_1(x_8).
\end{aligned}$$

where s_1, s_2, s_3 and s_4 are the 8×8 boxes.

$$\begin{aligned}
&P : (F_2^8)^8 \longrightarrow (F_2^8)^8\\
&y_1|y_2|y_3|y_4|y_5|y_6|y_7|y_8 \mapsto z_1|z_2|z_3|z_4|z_5|z_6|z_7|z_8
\end{aligned}$$

where

$$\begin{aligned}
z_1 &= y_1 \oplus y_3 \oplus y_4 \oplus y_6 \oplus y_7 \oplus y_8,\\
z_2 &= y_1 \oplus y_2 \oplus y_4 \oplus y_5 \oplus y_7 \oplus y_8,\\
z_3 &= y_1 \oplus y_2 \oplus y_3 \oplus y_5 \oplus y_6 \oplus y_8,\\
z_4 &= y_2 \oplus y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7,\\
z_5 &= y_1 \oplus y_2 \oplus y_6 \oplus y_7 \oplus y_8,\\
z_6 &= y_2 \oplus y_3 \oplus y_5 \oplus y_7 \oplus y_8,\\
z_7 &= y_3 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_8,\\
z_8 &= y_1 \oplus y_4 \oplus y_5 \oplus y_6 \oplus y_7.
\end{aligned}$$

The inverse of P is as follows:

$$\begin{aligned}
&P^{-1} : (F_2^8)^8 \longrightarrow (F_2^8)^8\\
&z_1|z_2|z_3|z_4|z_5|z_6|z_7|z_8 \mapsto y_1|y_2|y_3|y_4|y_5|y_6|y_7|y_8\\
&y_1 = z_2 \oplus z_3 \oplus z_4 \oplus z_6 \oplus z_7 \oplus z_8,\\
&y_2 = z_1 \oplus z_3 \oplus z_4 \oplus z_5 \oplus z_7 \oplus z_8,\\
&y_3 = z_1 \oplus z_2 \oplus z_4 \oplus z_5 \oplus z_6 \oplus z_8,\\
&y_4 = z_1 \oplus z_2 \oplus z_3 \oplus z_5 \oplus z_6 \oplus z_7,\\
&y_5 = z_1 \oplus z_2 \oplus z_5 \oplus z_7 \oplus z_8,\\
&y_6 = z_2 \oplus z_3 \oplus z_5 \oplus z_6 \oplus z_8,\\
&y_7 = z_3 \oplus z_4 \oplus z_5 \oplus z_6 \oplus z_7,\\
&y_8 = z_1 \oplus z_4 \oplus z_6 \oplus z_7 \oplus z_8.
\end{aligned}$$

3 Impossible Differential Cryptanalysis on Reduced-Round ARIA

3.1 Some 4-Round Impossible Differentials

In this subsection, we indicate some impossible differentials on 4-round ARIA as shown in Fig.1. In this figure, we consider the 4-round impossible differential which is built in a miss-in-the-middle manner. A 2-round differential with probability 1 is *concatenated* to a 2-round differential with probability 1, in the inverse direction, where the intermediate differences contradict each other. The 4-round impossible differential is $(a|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0) \xrightarrow{\text{4-round}} (0|h|0|0|0|0|0|0|h|h|h|0|0|0|h|0)$, where a and h denote any non-zero value.

We use $X_i^{\rm I}$ and $X_i^{\rm O}$ to denote the input and output of round i, while $X_i^{\rm S}$ denotes the intermediate values after the application of Substitution Layer (SL) of round i. The first 2-round differential is obtained as follows:

The input difference $X_1^{\rm I} = (a|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0)$ is preserved through the AddRoundKey operation of round 1. This difference is in a single byte, and thus, the difference after the Substitution Layer (SL) of round 1 is still in a single byte, i.e., $X_1^{\rm S} = (b|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0)$ where b is an unknown non-zero byte value. After the Diffusion Layer (DL) this difference becomes $X_2^{\rm I} = (0|0|0|b|b|0|b|0|b|b|0|0|0|b|b|0)$. This difference evolves after AddRoundKey operation and the Substitution Layer (SL) of round 2 into

$$X_2^{\rm S} = (0|0|0|b_3|b_4|0|b_6|0|b_8|b_9|0|0|0|b_{13}|b_{14}|0),$$

where $b_3, b_4, b_6, b_8, b_9, b_{13}$ and b_{14} are unknown non-zero byte values. Finally, after the Diffusion Layer (DL) this difference evolves to $X_2^{\rm O} = (c_0|c_1|c_2|c_3|\ldots|c_{15})$, where each byte can be expressed as:

$$\begin{aligned}
&c_0 = b_3 \oplus b_4 \oplus b_6 \oplus b_8 \oplus b_9 \oplus b_{13} \oplus b_{14}, \quad c_1 = b_8 \oplus b_9,\\
&c_2 = b_4 \oplus b_6, \quad c_3 = b_{13} \oplus b_{14},\\
&c_4 = b_8 \oplus b_{14}, \quad c_5 = b_3 \oplus b_4 \oplus b_9 \oplus b_{14},\\
&c_6 = b_9 \oplus b_{13}, \quad c_7 = b_3 \oplus b_6 \oplus b_8 \oplus b_{13},\\
&c_8 = b_4 \oplus b_{13}, \quad c_9 = b_6 \oplus b_{14},\\
&c_{10} = b_3 \oplus b_6 \oplus b_8 \oplus b_{13}, \quad c_{11} = b_3 \oplus b_4 \oplus b_9 \oplus b_{14},\\
&c_{12} = b_6 \oplus b_9, \quad c_{13} = b_3 \oplus b_6 \oplus b_8 \oplus b_{13},\\
&c_{14} = b_3 \oplus b_4 \oplus b_9 \oplus b_{14}, \quad c_{15} = b_4 \oplus b_8.
\end{aligned}$$

From the above equations, we get

$$\begin{aligned}
&c_7 = c_{10} = c_{13} = b_3 \oplus b_6 \oplus b_8 \oplus b_{13},\\
&c_{11} = c_{14} = b_3 \oplus b_4 \oplus b_9 \oplus b_{14}.
\end{aligned}$$

Hence, the input difference $X_1^{\rm I} = (a|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0)$ evolves with probability one into $X_2^{\rm O}$ which has the same value in bytes 11 and 14, and $X_2^{\rm O}$ also has the same value in bytes 7, 10, and 13.

The second differential ends after round 4 with difference $X_4^{\rm O} = (0|h|0|0|0|0|0|0|h|h|h|0|0|0|h|0)$. When rolling back this difference through the Diffusion Layer (DL), we get the difference $X_4^{\rm S} = (h|0|0|0|0|0|0|0|0|0|h|0|0|0|0|h)$. This difference has non-zero difference in bytes 0, 10, and 15, thus the difference evolves after the inverse of Substitution Layer (SL) and AddRoundKey operation of round 4 into $X_4^{\rm I} = (f_0|0|0|0|0|0|0|0|0|0|f_{10}|0|0|0|0|f_{15})$ where f_0, f_{10} and f_{15} are unknown non-zero byte values. When rolling back this difference through the Diffusion Layer (DL), we get the difference $X_3^{\rm S} = (e_0|e_1|e_2|e_3|e_4|e_5|e_6|e_7|e_8|e_9|e_{10}|e_{11}|e_{12}|e_{13}|e_{14}|e_{15})$, where each byte can be expressed as:

$$\begin{aligned}
&e_0 = 0, && e_8 = f_0 \oplus f_{10} \oplus f_{15},\\
&e_1 = f_{15}, && e_9 = f_0,\\
&e_2 = f_{10} \oplus f_{15}, && e_{10} = f_{15},\\
&e_3 = f_0 \oplus f_{10}, && e_{11} = 0,\\
&e_4 = f_0 \oplus f_{15}, && e_{12} = 0,\\
&e_5 = f_{10} \oplus f_{15}, && e_{13} = f_0 \oplus f_{10},\\
&e_6 = f_0 \oplus f_{10}, && c_{14} = f_0,\\
&e_7 = 0, && e_{15} = f_{10} \oplus f_{15}.
\end{aligned}$$

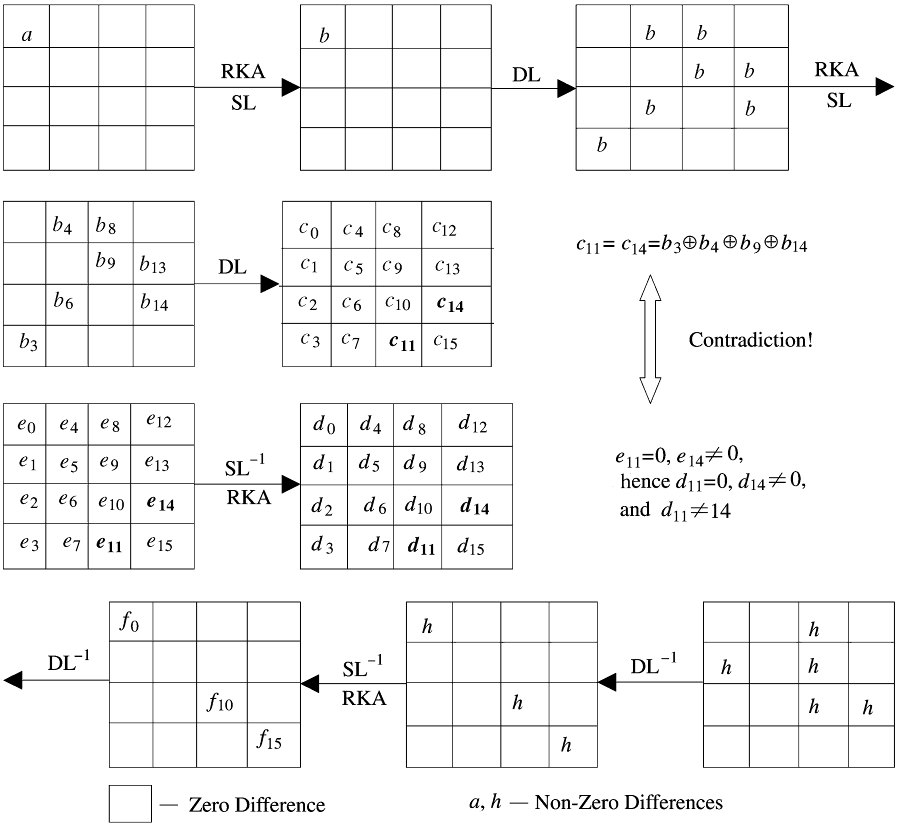


Fig.1. 4-Round impossible differentials of ARIA.

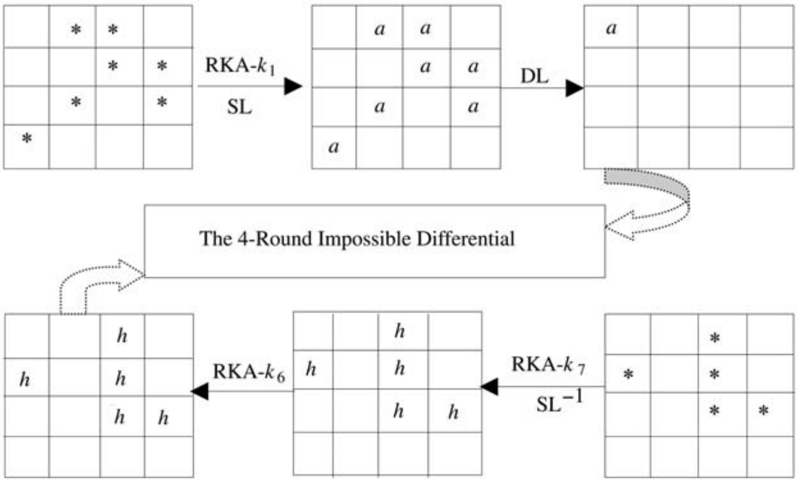


Fig.2. 6-round impossible differential attack to ARIA.

From the above equations, we know $e_{11} = 0$ and $e_{14} = f_0 \neq 0$. Therefore, when rolling back this difference through the inverse of Substitution Layer (SL) and AddRoundKey operation of round 3, we get the difference $X_3^{\mathrm{I}} = (d_0|d_1|d_2|d_3|d_4|d_5|d_6|d_7|d_8|d_9|d_{10}|d_{11}|d_{12}|d_{13}|d_{14}|d_{15})$, where $d_{11} = 0$ and $d_{14} \neq 0$.

This difference contradicts the first differential as with probability one $c_{11} = c_{14}$ while the second differential predicts $d_{11} \neq d_{14}$ with probability one. This contradiction is emphasized in Fig.1.

Similarly, we can get other 4-round impossible differentials of ARIA, for example,

$$(a|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0)$$
$$\xrightarrow{\text{4-round}}(0|0|h|0|h|0|0|0|0|0|0|0|h|h|h|0),$$
$$(a|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0)$$
$$\xrightarrow{\text{4-round}}(0|h|0|0|0|0|0|0|0|h|0|0|0|h|h|h),$$
$$(a|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0)$$
$$\xrightarrow{\text{4-round}}(0|0|0|0|h|0|0|h|h|0|0|0|h|0|h|0).$$

3.2 6-Round Impossible Differential Attack

In this subsection, we describe an impossible differential cryptanalysis of ARIA reduced to six rounds. The attack is based on the above four round impossible differentials with additional one round at each of the beginning and the end as in Fig.2. Note that the last round of ARIA does not have the diffusion layer.

Step 1. Choose structures of 2^{56} plaintexts which differ only at the seven bytes $(3, 4, 6, 8, 9, 13, 14)$, having all possible values in these bytes. One structure proposes $2^{56} \times 2^{56} \times \frac{1}{2} = 2^{111}$ pairs of plaintexts.

Step 2. Take 2^{64} structures (2^{120} plaintexts, 2^{175} pairs of plaintexts). Choose pairs whose ciphertext pairs have zero difference at the eleven bytes (0, 2, 3, 4, 5, 6, 7, 11, 12, 13, 15). The expected number of such pairs is about $2^{175} \times 2^{-88} = 2^{87}$.

Step 3. Guess the 40-bit value of the last round key k_7 at the five bytes (1, 8, 9, 10, 14), and perform the followings:

Step 3.1. For every remaining ciphertext pair (C, C^*), compute $C_5 \oplus C_5^* = SL^{-1}(C \oplus k_7) \oplus SL^{-1}(C^* \oplus k_7)$, choose pairs whose difference $C_5 \oplus C_5^*$ is the same at the five bytes (1, 8, 9, 10, 14). Since the probability is about $p = (2^{-8})^4 = 2^{-32}$, the expected number of the remaining pairs is about $2^{87} \times 2^{-32} = 2^{55}$.

Step 3.2. For every remaining ciphertext pair (C, C^*) consider the corresponding plaintext pair (P, P^*), for 56-bit value at the seven bytes (3, 4, 6, 8, 9, 13, 14) of the subkey k_1, calculate $SL(P \oplus k_1) \oplus SL(P^* \oplus k_1)$, and check whether $SL(P \oplus k_1) \oplus SL(P^* \oplus k_1)$ is the same at the seven bytes (3, 4, 6, 8, 9, 13, 14). If yes, discard the candidate value of the seven bytes of k_1 and the five bytes of k_7.

The procedure is as follows.

Since such a difference is impossible, every key that proposes such a difference is a wrong key. After analyzing 2^{55} ciphertext pairs, there remain only about $2^{56}(1 - 2^{-48})^{2^{55}} \approx 2^{56}e^{-2^7} \approx 2^{-128}$ wrong values of the seven bytes of k_1. Unless the initial assumption on the five bytes of k_7 is right, it is expected that we can detect the whole 56-bit value of k_1 for each 40-bit value of k_7 since the wrong value remains with the probability 2^{-88}. Hence if there remains a value of k_1, we can assume the value k_7 is right.

The time complexity of the attack is dominated by Step 3. For reducing the time complexity of Step 3.1, we first compute $C_{(5,1)} \oplus C^*_{(5,1)}$ and $C_{(5,8)} \oplus C^*_{(5,8)}$, and check whether $C_{(5,1)} \oplus C^*_{(5,1)} = C_{(5,8)} \oplus C^*_{(5,8)}$, it needs only to guess two key bytes. If yes, go on computing $C_{(5,9)} \oplus C^*_{(5,9)}$, and so on. Thus Step 3.1 requires about $4 \times 2^{103} = (2^{16} \times 2^{87} + 2^{24} \times 2^{79} + 2^{32} \times 2^{71} + 2^{40} \times 2^{63})$ one round operations. Step 3.2 requires about $6 \times 2^{111} (= 2^{40} \times (2^{16} \times 2^{55} + 2^{24} \times 2^{47} + \cdots + 2^{56} \times 2^{15})$, one round operations.

Similarly, we can derive the other bytes of k_7 by using different impossible differentials. Consequently, this attack requires about 2^{121} chosen plaintexts and 2^{112} encryptions of 6-round ARIA.

4 Impossible Differential Cryptanalysis on Reduced-Round Camellia

4.1 Some 8-Round Impossible Differentials

In [4], the authors show one impossible differential of 7-round Camellia without input/output whitening, FL, or FL^{-1}. In this subsection, we indicate one impossible differential of 8-round Camellia as shown in Fig.3.

We now show the 8-round differential $(0|0|0|0|0|0|0|0, a|0|0|0|0|0|0|0) \xrightarrow{\text{8-round}} (h|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|0)$ is impossible, where a and h denote any non-zero value.

The first 3-round differential is obtained as follows:

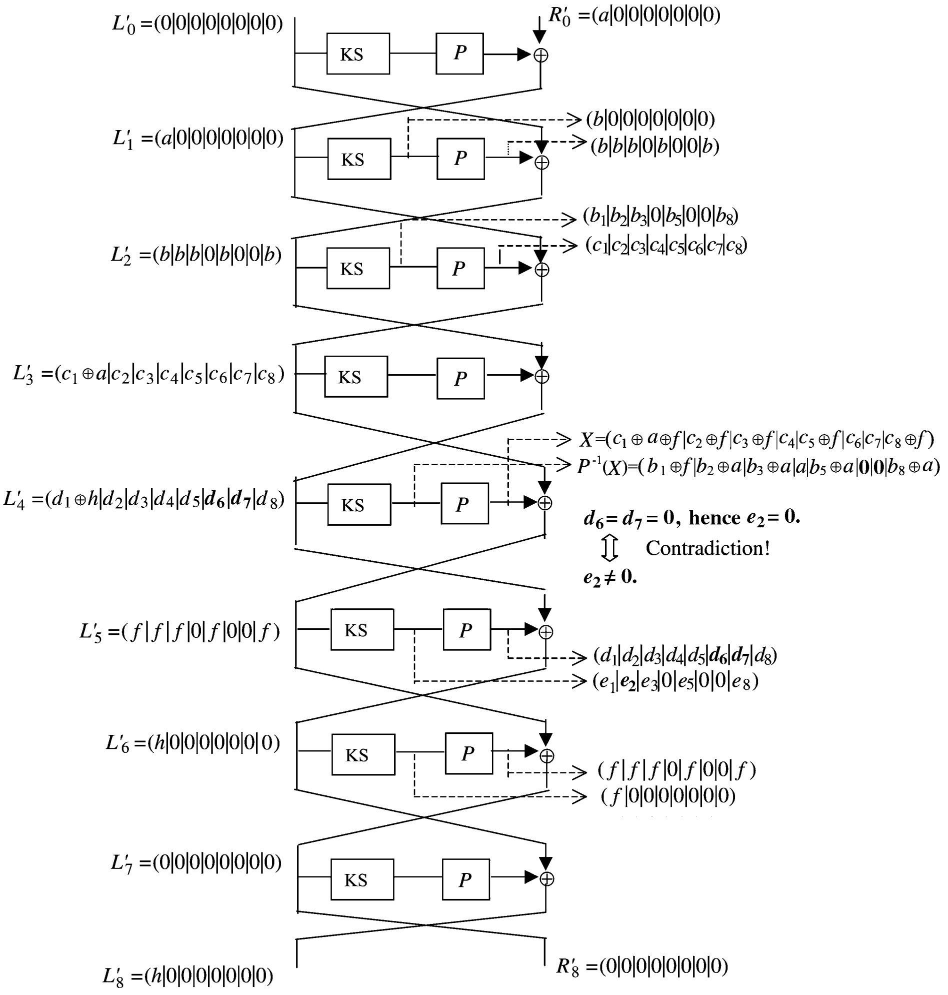
The input difference $(L'_0, R'_0) = (0|0|0|0|0|0|0|0, a|0|0|0|0|0|0|0)$ becomes $(L'_1, R'_1) = (a|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|0)$ through the first round transformation. After the subkey addition and S layer, L'_1 becomes $(b|0|0|0|0|0|0|0)$ where b is an unknown non-zero byte value. After the linear transformation P we have $(L'_2, R'_2) = (b|b|b|0|b|0|0|b, a|0|0|0|0|0|0|0)$. This difference evolves after subkey addition operation and the S-box layer of round 3 into $(b_1|b_2|b_3|0|b_5|0|0|b_8)$, where b_1, b_2, b_3, b_5 and b_8 are unknown non-zero byte values. Further, after the linear transformation P this difference evolves to $(c_1|c_2|c_3|c_4|c_5|c_6|c_7|c_8)$. Thus we get $(L'_3, R'_3) = (c_1 \oplus a|c_2|c_3|c_4|c_5|c_6|c_7|c_8, b|b|b|0|b|0|0|b)$.

The second 3-round differential ends with difference $(L'_8, R'_8) = (h|0|0|0|0|0|0|0, 0|0|0|0|0|0|0|0)$. When rolling back this difference through 2-round transformation, we get the difference $(L'_6, R'_6) = (h|0|0|0|0|0|0|0, f|f|f|0|f|0|0|f)$, where f is an unknown non-zero byte value. After the subkey addition and S layer, $L'_5 = R'_6$ becomes $(e_1|e_2|e_3|0|e_5|0|0|e_8)$, where e_1, e_2, e_3, e_5 and e_8 are unknown non-zero byte values. Further, after the linear transformation P this difference evolves to $(d_1|d_2|d_3|d_4|d_5|d_6|d_7|d_8)$. Thus we get $(L'_5, R'_5) = (f|f|f|0|f|0|0|f, d_1 \oplus h|d_2|d_3|d_4|d_5|d_6|d_7|d_8)$, where d_6 and d_7 can be expressed as:

$$d_6 = e_2 \oplus e_3 \oplus e_5 \oplus e_8, \quad d_7 = e_3 \oplus e_5 \oplus e_8.$$

If the first 3-round differential and second 3-round differential can build up the 8-round differential, then L_3, L_5 and R_5 must satisfy the following:

$$L_4 = R_5, \quad P(S(R_5 \oplus k_4)) = L_3 \oplus L_5.$$



L'_0 =(0|0|0|0|0|0|0|0)
R'_0 =(a|0|0|0|0|0|0|0)
KS
P
L'_1 =(a|0|0|0|0|0|0|0)
(b|0|0|0|0|0|0|0)
(b|b|b|0|b|0|0|b)
L'_2 =(b|b|b|0|b|0|0|b)
(b_1|b_2|b_3|0|b_5|0|0|b_8)
(c_1|c_2|c_3|c_4|c_5|c_6|c_7|c_8)
L'_3 =($c_1 \oplus a$|c_2|c_3|c_4|c_5|c_6|c_7|c_8)
X=($c_1 \oplus a \oplus f$|$c_2 \oplus f$|$c_3 \oplus f$|c_4|$c_5 \oplus f$|c_6|c_7|$c_8 \oplus f$)
$P^{-1}(X)$=($b_1 \oplus f$|$b_2 \oplus a$|$b_3 \oplus a$|a|$b_5 \oplus a$|0|0|$b_8 \oplus a$)
L'_4 =($d_1 \oplus h$|d_2|d_3|d_4|d_5|d_6|d_7|d_8)
$d_6 = d_7 = 0$, hence $e_2 = 0$.
⇕ Contradiction!
$e_2 \neq 0$.
L'_5 =(f|f|f|0|f|0|0|f)
(d_1|d_2|d_3|d_4|d_5|d_6|d_7|d_8)
(e_1|e_2|e_3|0|e_5|0|0|e_8)
L'_6 =(h|0|0|0|0|0|0|0)
(f|f|f|0|f|0|0|f)
(f|0|0|0|0|0|0|0)
L'_7 =(0|0|0|0|0|0|0|0)
L'_8 =(h|0|0|0|0|0|0|0)
R'_8 =(0|0|0|0|0|0|0|0)

Fig.3. 8-round impossible differentials of Camellia.

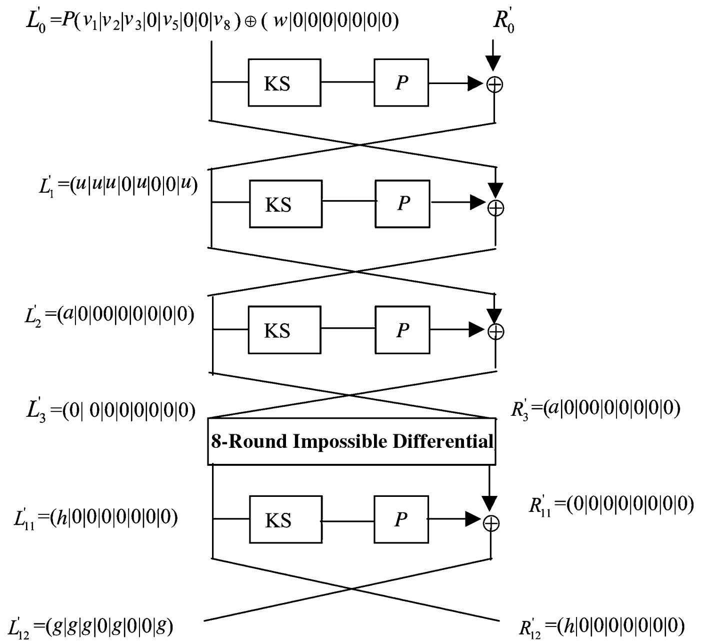


Fig.4. 12-round impossible differential attack to modified Camellia.

Hence we have $S(R_5 \oplus k_4) = P^{-1}(L_3 \oplus L_5)$. Because P^{-1} is a linear transformation, we have

$$\begin{aligned}
&P^{-1}(L'_3 \oplus L'_5) = P^{-1}(L'_3) \oplus P^{-1}(L'_5)\\
&= P^{-1}(c_1 \oplus a|c_2|c_3|c_4|c_5|c_6|c_7|c_8) \oplus P^{-1}(f|f|f|0|f|0|0|f)\\
&= P^{-1}(c_1|c_2|c_3|c_4|c_5|c_6|c_7|c_8)\\
&\quad \oplus P^{-1}(a|0|0|0|0|0|0|0) \oplus P^{-1}(f|f|f|0|f|0|0|f)\\
&= (b_1|b_2|b_3|0|b_5|0|0|b_8) \oplus (0|a|a|a|a|0|0|a)\\
&\quad \oplus (f|0|0|0|0|0|0|0)\\
&= (f \oplus b_1|b_2 \oplus a|b_3 \oplus a|a|b_5 \oplus a|0|0|b_8 \oplus a).
\end{aligned}$$

The S-boxes of Camellia are permutations, so we can get the sixth and seventh byte difference in R'_5 equals zero, i.e., $d_6 = d_7 = 0$. From the expression of d_6 and d_7 we have $d_6 \oplus d_7 = e_2$. This contradicts with $e_2 \neq 0$.

Similarly, we can get other 8-round impossible differentials of Camellia, for example,

$$\begin{gathered}
(0|0|0|0|0|0|0|0, 0|a|0|0|0|0|0|0)\\
\xrightarrow{\text{8-round}}(0|h|0|0|0|0|0|0, 0|0|0|0|0|0|0|0),\\
(0|0|0|0|0|0|0|0, 0|0|a|0|0|0|0|0)\\
\xrightarrow{\text{8-round}}(0|0|h|0|0|0|0|0, 0|0|0|0|0|0|0|0).
\end{gathered}$$

4.2 12-Round Impossible Differential Attack

In this subsection, we describe an impossible differential attack of 12-round Camellia without whitening and FL/FL^{-1}. The attack is based on the above 8-round impossible differentials with additional three rounds at the beginning and one round at the end as in Fig.4.

The procedure is as follows:

Step 1. Choose structure of plaintexts as follows:

$$\begin{aligned}
L_0 &= P(x_1|x_2|x_3|\alpha_4|x_5|\alpha_6|\alpha_7|x_8)\\
&\quad \oplus (x|\beta_2|\beta_3|\beta_4|\beta_5|\beta_6|\beta_7|\beta_8),\\
R_0 &= (y_1|y_2|y_3|y_4|y_5|y_6|y_7|y_8).
\end{aligned}$$

where x_i $(i = 1, 2, 3, 5, 8)$, y_i $(1 \leqslant i \leqslant 8)$, and x takes all possible values in F_2^8, α_i and β_i are constants in F_2^8. For each possible value of $(x_1, x_2, x_3, x_5, x_8, x, y_1, \ldots, y_8)$, we can get a unique 128-bit string $(P(x_1|x_2|x_3|\alpha_4|x_5|\alpha_6|\alpha_7|x_8) \oplus (x|\beta_2|\beta_3|\beta_4|\beta_5|\ \beta_6|\beta_7|\beta_8), (y_1|y_2|y_3|y_4|y_5|y_6|y_7|y_8))$. Also, for a different value of $(x_1, x_2, x_3, x_5, x_8, x, y_1, \ldots, y_8)$, the corresponding 128-bit string is also different. Hence, a structure includes 2^{112} plaintexts, one structure proposes $2^{112} \times 2^{112} \times \frac{1}{2} = 2^{223}$ pairs of plaintexts.

Step 2. Take 2^8 structures (2^{120} plaintexts, 2^{231} pairs of plaintexts). Choose pairs whose ciphertext difference (L'_{12}, R'_{12}) satisfy the following:

$$L'_{12} = (g|g|g|0|g|0|0|g), \quad R'_{12} = (h|0|0|0|0|0|0|0),$$

where h and g are unknown non-zero values. There are 2^{16} (L'_{12}, R'_{12}), so the probability is about $p = 2^{16} \times 2^{-128} = 2^{-112}$. Hence, the expected number of such pairs is $2^{231} \times 2^{-112} = 2^{119}$.

Step 3. Guess the 8-bit value at the first byte of the subkey k_{12}, for every remaining pair, calculate $s_1(R_{12,1} \oplus k_{12,1}) \oplus s_1(R^*_{12,1} \oplus k_{12,1})$, and choose pairs which satisfy $s_1(R_{12,1} \oplus k_{12,1}) \oplus s_1(R^*_{12,1} \oplus k_{12,1}) = L_{12,1} \oplus L^*_{12,1}$. Since the probability is about $p = 2^{-8}$, the expected number of the remaining pairs is $2^{119} \times 2^{-8} = 2^{111}$.

Step 4. Guess the 64-bit value of the first round key k_1, for every remaining plaintext pair (L_0, R_0) and (L^*_0, R^*_0),

$$\begin{aligned}
L_0 &= P(x_1|x_2|x_3|\alpha_4|x_5|\alpha_6|\alpha_7|x_8)\\
&\quad \oplus (x|\beta_2|\beta_3|\beta_4|\beta_5|\beta_6|\beta_7|\beta_8),\\
R_0 &= (y_1|y_2|y_3|y_4|y_5|y_6|y_7|y_8),\\
L^*_0 &= P(x^*_1|x^*_2|x^*_3|\alpha_4|x^*_5|\alpha_6|\alpha_7|x^*_8)\\
&\quad \oplus (x^*|\beta_2|\beta_3|\beta_4|\beta_5|\beta_6|\beta_7|\beta_8),\\
R^*_0 &= (y^*_1|y^*_2|y^*_3|y^*_4|y^*_5|y^*_6|y^*_7|y^*_8).
\end{aligned}$$

Compute (L_1, R_1) and (L^*_1, R^*_1), and choose pairs whose difference satisfies $L_1 \oplus L^*_1 = (u|u|u|0|u|0|0|u)$ where u is not zero. Since the probability is about $p = 2^8 \times 2^{-64} = 2^{-56}$, the expected number of the remaining pairs is $2^{111} \times 2^{-56} = 2^{55}$.

Step 5. Guess the 40-bit value of the second round key k_2 at the five bytes (1, 2, 3, 5, 8), perform the following:

Step 5.1. For every remaining pair (L_0, R_0) and (L_0^*, R_0^*), and the corresponding output of the first round (L_1, R_1) and (L_1^*, R_1^*),

$$\begin{aligned}
L_1 &= (z_1|z_2|z_3|\gamma_4|z_5|\gamma_6|\gamma_7|z_8),\\
R_1 &= P(x_1|x_2|x_3|\alpha_4|x_5|\alpha_6|\alpha_7|x_8) \oplus (x|\beta_2|\beta_3|\beta_4|\beta_5|\beta_6|\beta_7|\beta_8),\\
L_1^* &= (z_1^*|z_2^*|z_3^*|\gamma_4|z_5^*|\gamma_6|\gamma_7|z_8^*),\\
R_1^* &= P(x_1^*|x_2^*|x_3^*|\alpha_4|x_5^*|\alpha_6|\alpha_7|x_8^*) \oplus (x^*|\beta_2|\beta_3|\beta_4|\beta_5|\beta_6|\beta_7|\beta_8).
\end{aligned}$$

Compute $s_1(z_1 \oplus k_{2,1}) \oplus s_1(z_1^* \oplus k_{2,1}) = v_1$, $s_2(z_2 \oplus k_{2,2}) \oplus s_2(z_2^* \oplus k_{2,2}) = v_2$, $s_3(z_3 \oplus k_{2,3}) \oplus s_3(z_3^* \oplus k_{2,3}) = v_3$, $s_2(z_5 \oplus k_{2,5}) \oplus s_2(z_5^* \oplus k_{2,5}) = v_5$, $s_1(z_8 \oplus k_{2,8}) \oplus s_1(z_8^* \oplus k_{2,8}) = v_8$. Choose pairs whose difference satisfy $(v_1|v_2|v_3|v_5|v_8) = (x_1 \oplus x_1^*|x_2 \oplus x_2^*|x_3 \oplus x_3^*|x_5 \oplus x_5^*|x_8 \oplus x_8^*)$ and $x \neq x^*$. Since the probability is about $p = 2^{-40}$, the expected number of the remaining pairs is $2^{55} \times 2^{-40} = 2^{15}$.

Step 5.2. Further guess the 24-bit value of the second round key k_2 at the three bytes (4, 6, 7), for every remaining plaintext pair, calculate $L_{2,1}$ and $L_{2,1}^*$.

Step 6. For 8-bit value at the first byte of the subkey k_3, for every remaining plaintext pair, calculate $s_1(L_{2,1} \oplus k_{3,1}) \oplus s_1(L_{2,1}^* \oplus k_{3,1})$, and check whether $s_1(L_{2,1} \oplus k_{3,1}) \oplus s_1(L_{2,1}^* \oplus k_{3,1}) = L_{1,1} \oplus L_{1,1}^*$. If yes, discard the candidate value of $(k_1, k_2, k_{3,1}, k_{12,1})$.

Since such a difference is impossible, every key that proposes such a difference is a wrong key. After analyzing 2^{15} ciphertext pairs, there remain only about $2^{144}(1 - 2^{-8})^{2^{15}} \approx 2^{144}e^{-2^7} \approx 2^{-50}$ wrong candidate value of $(k_1, k_{2,1}, k_{11,1}, k_{12})$.

In Step 2, the cost to check all the possible pairs of ciphertext in 2^8 structure require 2^{231}, that is larger than the value of 2^{192}. However, this procedure does not require any encryption function. So the complexity can be ignored.

The time complexity of Step 3 requires about $2^{127} = 2^8 \times 2^{119}$ one-round operations. Step 4 requires about $2^{183} = 2^{64} \times 2^8 \times 2^{111}$ one-round operations. Step 5.1 requires about $2^{167} = 2^8 \times 2^{64} \times 2^{40} \times 2^{55}$ one-round operations. Step 5.2 requires about $2^{151} = 2^8 \times 2^{64} \times 2^{64} \times 2^{15}$ one-round operations. Step 6 requires about $2^{159} = 2^{72} \times 2^{72} \times 2^{15}$ one-round operations.

Consequently, this attack requires about 2^{120} chosen plaintexts and less than 2^{181} encryptions of 12-round Camellia.

5 Concluding Remarks

In this paper, we examine the security of ARIA and Camellia against impossible differential attacks. The designers of ARIA expected no impossible differentials existing on 4-round ARIA. However, we found some nontrivial 4-round impossible differentials, and then presented an attack to 6-round ARIA with data complexity 2^{121} and 2^{112} encryptions. As for Camellia, we found some nontrivial 8-round impossible differentials for Camellia, whereas only 7-round impossible differentials were previously known. By using the 8-round impossible differential, we presented an attack on 12-round Camellia with data complexity 2^{120} and 2^{181} encryptions, the attack having complexity less than that of exhaustive search to 12-round Camellia without FL/FL^{-1} layers.

Since ARIA is a new cipher published in 2004, all we know about its security is limited to the designers' analysis and that of [11]. Here we only compare the complexities of our attack with those of previous work on Camellia in Table 1.

Table 1. Summary of Known Attacks on Camellia

Rounds	FL/FL^{-1}	Methods	Data	Time	Notes
7	×	Impossible DC	–	–	Ref.[4]
8	×	Truncated DC	$2^{83.6}$	$2^{55.6}$	Ref.[3] (128-bit key)
9	√	Boomerang	2^{124}	2^{170}	Ref.[7] (192/256-bit key)
9	×	Collision Attack	2^{13}	$2^{175.6}$	Ref.[9] (192/256-bit key)
9	√	Integral Attack	$2^{60.5}$	$2^{202.2}$	Ref.[8] (256-bit key)
9	√	Square Attack	2^{60}	2^{202}	Ref.[6] (256-bit key)
10	√	Rectangle	2^{127}	2^{241}	Ref.[7] (256-bit key)
10	×	Collision Attack	2^{14}	$2^{239.9}$	Ref.[9] (256-bit key)
10	×	Variant Square Attack	–	2^{186}	Ref.[10] (192/256-bit key)
11	×	DC	2^{104}	2^{232}	Ref.[7] (256-bit key)
11	×	Variant Square Attack	–	2^{250}	Ref.[10] (256-bit key)
11	×	Higher Order DC	2^{21}	2^{255}	Ref.[5] (256-bit key)
11	√	Higher Order DC	2^{93}	2^{256}	Ref.[5] (256-bit key)
12	×	Linear Attack	2^{119}	2^{247}	Ref.[7] (256-bit key)
12	×	Impossible DC	2^{120}	2^{181}	This paper (192/256-bit key)

References

[1] Daesung Kwon, Jaesung Kim, Sangwoo Park *et al.* New block cipher: ARIA. In *Proc. Information Security and Cryptology* (*ICISC'03*), Seoul, Korea, *LNCS* 2971, Springer-Verlag, November 27∼28, 2003, pp.432∼445.

[2] Aoki K, Ichikawa T, Kanda M *et al.* Specification of Camellia — A 128-bit block cipher. In *Proc. Selected Areas in Cryptography* (*SAC'2000*), Waterloo, Canada, *LNCS* 2012, Springer-Verlag, August 14∼15, 2000, pp.183∼191.

[3] Lee S, Hong S, Lee S *et al.* Truncated differential cryptanalysis of Camellia. In *Proc. Information Security and Cryptology* (*ICISC'01*), Seoul, Korea, *LNCS* 2288, Springer-Verlag, December 6∼7, 2001, pp.32∼38.

[4] Sugita M, Kobara K, Imai H. Security of reduced version of the block cipher Camellia against truncated and impossible differential cryptanalysis. In *Proc. Advances in Cryptology* (*Asiacrypt'01*), Queensland, Australia, *LNCS* 2248, Springer-Verlag, December 9∼13, 2001, pp193∼207.

[5] Hatano Y, Sekine H, Kaneko T. Higher order differential attack of Camellia (II). In *Proc. Selected Areas in Cryptography* (*SAC'02*), Newfoundland, Canada, *LNCS* 2595, Springer-Verlag, August 15∼16, 2002, pp.39∼56.

[6] Yeom Y, Park S, Kim I. On the security of Camellia against the square attack. In *Proc. Fast Software Encryption* (*FSE'02*), Springer-Verlag, Leuven, Belgium, *LNCS* 2356, February 4∼6, 2002, pp.89∼99.

[7] Shirai T. Differential, linear, boomerang and rectangle cryptanalysis of reduced-round Camellia. In *Proc. the Third NESSIE Workshop*, Munich, Germany, November 6∼7, 2002. Available at: https://www.cosic.esat.kuleuven.be/nessie/.

[8] Yeom Y, Park I, Kim I. A study of integral type cryptanalysis on Camellia. In *Proc. The 2003 Symposium on Cryptography and Information Security* (*SCIS'03*), Hamamatsu, Japan, January 2003, pp.26∼29.

[9] Wenling Wu, Dengguo Feng, Hua Chen. Collision attack and pseudorandomness of reduced-round Camellia. In *Proc. Selected Areas in Cryptography* (*SAC 2004*), Waterloo, Canada, *LNCS* 3357, Springer-Verlag, August 9∼10, 2004, pp.256∼270.

[10] Duo Lei, Li Chao, Keqin Feng. New observation on Camellia. In *Proc. Selected Areas in Cryptography* (*SAC 2005*), Springer-Verlag, Kingston, Canada, *LNCS* 3897, August 11∼12, 2005, pp.51∼64.

[11] Wenling Wu. Pseudorandomness of Camellia-like scheme. *Journal of Computer Science and Technology*, 2006, 21(1): 82∼88.

[12] A Biryukov, Christophe De Canniere *et al.* Security and performance analysis of ARIA. Available at http://homes.esat.kuleuven.be/∼abiryuko/ARIA-COSICreport.pdf.

[13] Biham E, Shamir A. Differential Cryptanalysis of the Data Encryption Standard, Springer-Verlag, 1993.

[14] Matsui M. Linear cryptanalysis method for DES cipher. In *Proc. Advances in Cryptology–EUROCRYPT'93*, Lofthus, Norway, *LNCS* 765, Springer-Verlag, May 23∼27, 1993, pp.386∼397.

[15] Knudsen L. Truncated and higher order differentials. In *Proc. Fast Software Encryption* (*FSE'95*), Leuven, Belgium, *LNCS* 2595, Springer-Verlag, December 1994, pp.196∼211.

[16] Biham E, Biryukov A, Shamir A. Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. In *Proc. Advances in Cryptology–EUROCRYPT'99*, Rague, Czech Republic, *LNCS* 2595, Springer-Verlag, May 2∼6, 1999, pp.12∼23.

[17] Biryukov A, Wagner D. Slide attacks. In *Proc. Fast Software Encryption* (*FSE'99*), Rome, Italy, *LNCS* 1636, Springer-Verlag, March 24∼26, 1999, pp.245∼259.

[18] Biryukov A, Wagner D. Advanced slide attacks. In *Proc. Advances in Cryptology–EUROCRYPT'00*, Bruges, Belgium, *LNCS* 1807, Springer-Verlag, May 14∼18, 2000, pp.589∼606.

[19] Knudsen L, Wagner D. Integral cryptanalysis (extended abstract). In *Proc. Fast Software Encryption* (*FSE 2002*), Leuven, Belgium, *LNCS* 2595, Springer-Verlag, February 4∼6, 2002, pp.112∼127.

[20] Wagner D. The boomerang attack. In *Proc. Fast Software Encryption* (*FSE'99*), Rome, Italy, *LNCS* 1636, Springer-Verlag, March 24∼26, 1999, pp.157∼170.

[21] Jakobsen T, Knudsen L. The interpolation attack against block ciphers. In *Proc. Fast Software Encryption* (*FSE'99*), Rome, Italy, *LNCS* 1267, Springer-Verlag, pp.28∼40.

[22] Courtois N, Pieprzyk J. Cryptanalysis of block ciphers with overdefined systems of equations. In *Proc. Advances in Cryptology–ASIACRYPT'02*, Queenstown, New Zealand, *LNCS* 2595, Springer-Verlag, December 1∼5, 2002, pp.267∼287.

[23] Jung Hee Cheon, Munju Kim, Kwangjo Kim *et al.* Improved impossible differential cryptanalysis of Rijndael and Crypton. In *Proc. International Conference on Information Security and Cryptology* (*ICISC'01*), Seoul, South Korea, *LNCS* 2288, Springer-Verlag, December 6∼7, 2001, pp.39∼49.

[24] Raphael Chung-Wei Phan. Impossible differential cryptanalysis of 7-round AES. *Information Processing Letters*, 2004, 91(1): 33∼38.

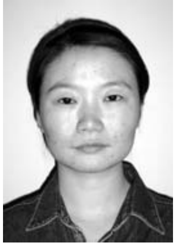
[25] Goce Jakimoski, Yvo Desmedt. Related-key differential cryptanalysis of 192-bit key AES variants. In *Proc. Selected Areas in Cryptography* (*SAC'2003*), Ottawa, Canada, *LNCS* 3006, Springer-Verlag, August 14∼15, 2003, pp.208∼221.

[26] Biham E, Orr Dunkelman, Nathan Keller. Related-key impossible differential attacks on 8-round AES-192. In *Proc. The Cryptographer's Track* (*CT-RSA*), San Jose, CA, USA, *LNCS* 3860, Springer-Verlag, February 13∼17, 2006, pp.21∼33.

[27] Wentao Zhang, Wenling Wu, Lei Zhang, Dengguo Feng. Improved related-key impossible differential attacks on reduced-round AES-192. In *Proc. Selected Areas in Cryptography* (*SAC'2006*), Montreal, Canada, Springer-Verlag, August 17∼18, 2006, pp.168∼181.

[28] Bon Wook Koo, Hwan Seok Jang, Jung Hwan Song. Constructing and cryptanalysis of a 16 × 16 binary matrix as a diffusion layer. In *Proc. Int. Workshop on Information Security Applications*, Jeju Island, Korea, *LNCS* 2908, Springer-Verlag, August 25∼27, 2003, pp.489∼503.

Wen-Ling Wu is now a professor and Ph.D. supervisor at the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences. She received her B.S. degree and M.S. degree in maths from Northwest University in 1987 and 1990, respectively. She received her Ph.D. degree in cryptography from Xidian University in 1997. From 1998 to 1999 she was a postdoctoral fellow in the Institute of Software, Chinese Academy of Science. She is a senior member of China Computer Federation. Her current research interests include theory of cryptography, mode of operation, block cipher, stream cipher and hash function.



Wen-Tao Zhang is now an assistant professor at the State Key Laboratory of Information Security, Graduate University of Chinese Academy of Sciences. She received her M.S. degree in maths from Northwest University in 2000. She received her Ph.D. degree in computer science and technology from Institute of Software, Chinese Academy of Sciences in 2003. Her current research interest is the design and analysis of block cipher.

Deng-Guo Feng is now a professor and Ph.D. supervisor at the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences. In 1995, he received his Ph.D. degree in communication and information system from Xidian University and began to work as a post doctoral fellow with the Graduate School of University of Science and Technology of China. In 1997, he joined IOS and was elected into the project of One Hundred Talents of CAS. He is a senior member of China Computer Federation. His research interests are in the areas of cryptology and information security.