A Meet-in-the-Middle Attack on ARIA

Xuehai Tang[1], Bing Sun[1], Ruilin Li[1] and Chao Li[1,2]

[1] Department of Mathematics and System Science, Science College of National University of Defense Technology, Changsha, China, 410073
[2] State Key Laboratory of Information Security, Graduate University of Chinese Academy of Sciences, China, 100190
{txh0203, happy_come}@163.com; securitylrl@gmail.com; lichao_nudt@sina.com

Abstract. In this paper, we study the meet-in-the-middle attack against block cipher ARIA. We find some new 3-round and 4-round distinguishing properties of ARIA. Based on the 3-round distinguishing property, we can apply the meet-in-the-middle attack with up to 6 rounds for all versions of ARIA. Based on the 4-round distinguishing property, we can mount a successful attack on 8-round ARIA-256. Furthermore, the 4-round distinguishing property could be improved which leads to a 7-round attack on ARIA-192. The data and time complexities of 7-round attack are 2^{120} and $2^{185.3}$, respectively. The data and time complexities of 8-round attack are 2^{56} and $2^{251.6}$, respectively. Compared with the existing cryptanalytic results on ARIA, our 5-round attack has the lowest data and time complexities and the 6-round attack has the lowest data complexity. Moreover, it is shown that 8-round ARIA-256 is not immune to the meet-in-the-middle attack.

Key words: block cipher, ARIA, meet-in-the-middle, time-memory trade-off

1 Introduction

ARIA[1] is a 128-bit block cipher designed by a group of Korean experts in 2003. Its design adopts the same idea(wide trail strategy) of the Advanced Encryption Standard(AES)[2]. It was later established as a Korean Standard by the Ministry of Commerce, Industry and Energy in 2004. ARIA supports key length of 128, 192 and 256 bits, these versions of ARIA are denoted as ARIA-128, ARIA-192 and ARIA-256. The number of rounds for these three versions are 12, 14 and 16, respectively.

The security of ARIA was analyzed by many cryptographists. In [1], the designers of ARIA presented some cryptoanalysis including both differential cryptanalysis, linear cryptanalysis, and some other known attacks . Later Biryukov *et al.* performed an evaluation of ARIA [3], however, they especially focused on truncated differential cryptanalysis and dedicated linear cryptanalysis. In ref. [4], Wu *et al.* firstly found some non-trivial 4-round impossible differentials which led to a 6-round attack on ARIA. Li *et al.* presented an algorithm to find

many new 4-round impossible differentials which can improve the 6-round impossible differential attack [5]. The security of ARIA against boomerang attack was presented by Fleischmann *et al.* in [6]. And recently, Li *et al.* firstly found some 3-round integral distinguishers by counting methods, which also led up to a 6-round integral attack on ARIA-192 [7].

The meet-in-the-middle attack on AES was firstly introduced by Demirci *et al.* in [8]. Inspired by their work, we construct some new 3/4-round distinguishing properties of ARIA and use them to apply the meet-in-the-middle attack against ARIA. Based on the 3-round distinguishing property, we can attack all versions of ARIA with up to 6 rounds. Based on the 4-round distinguishing property, we can mount a successful attack on 8-round ARIA-256. Furthermore, we improve the 4-round distinguishing property and use it to attack 7-round ARIA-192. Our results show that the 5-round attack has the lowest data and time complexities and the 6-round attack has the lowest data complexity compared with the existing attacks on ARIA. Although this kind of attack has a huge precomputation and memory complexity, the precomputation only needs to compute once. To validate the correctness of the meet-in-the-middle attack, we also do some experiments on 3-round ARIA.

The rest of this paper is organized as follows: We describe the meet-in-the-middle attack in Section 2 and give a brief description of ARIA in Section 3. In Section 4, we construct some 3/4-round distinguishing properties of ARIA and present the meet-in-the-middle attacks on the round-reduced ARIA. We do some experimental results of the meet-in-the-middle attack on 3-round ARIA in Section 5. Finally, Section 6 summarizes this paper.

2 The Meet-in-the-Middle Attack

The idea of meet-in-the-middle attack was firstly introduced by Diffie and Hellman in cryptanalysis of Two-DES [9], the main idea is using the technique of time-memory tradeoff. Demirci *et al.* extended the meet in the middle attack in a more generalized case and applied it to attack 8-round AES-256 [8, 10] based on some 5-round distinguishing property, which originates from an early 4-round distinguishing property [11] constructed by Gilbert and Minier.

In this section, we describe in detail the generalized meet-in-the-middle attack against iterative block ciphers.

Let an N-round block cipher be

$$C = E(P, K), \tag{1}$$

where C, P and K denote ciphertext, plaintext and the user key, respectively. The encryption procedure is treated as a concatenation of two consecutive encryptions, namely E_1 and E_2, i.e. $E = E_2 \circ E_1$, where E_1 is the first N_1 rounds encryption and E_2 the last $N_2 = N - N_1$ rounds encryption, thus

$$C = E_2(E_1(P, K_1), K_2), \tag{2}$$

where K_1 and K_2 are the subkeys of the first N_1 and the last N_2 rounds, respectively.

If we consider m different plaintexts with the feature that they are different at some fixed bits(denoted as x) only and the rest bits are constant values. Denote the m plaintexts as $x_1, x_2, \ldots, x_m$, encrypt the m palintexts with the first N_1 rounds, we can compute the ciphertexts $C_i^* = E(x_i, K_1)$, where $1 \leq i \leq m$. Usually, we consider a partial bits of C_i^* , denoted as c_i. Note that the constant values in the plaintexts and K_1 are fixed for each ciphertext c_i, then c_i can be expressed as the function with the variable x_i:

$$c_i = f(x_i) \tag{3}$$

where f is determined by some parameters and the subkey K_1 is included in the parameters. If the number of parameters in $f(x)$ is small enough, we can search exhaustively all the parameters and the right subkey K_1 must be included. In other words, for each possible parameter, we can obtain a mapping $f(x_i) : x_i \rightarrow c_i$, thus we can obtain many mappings and only one mapping is correct.

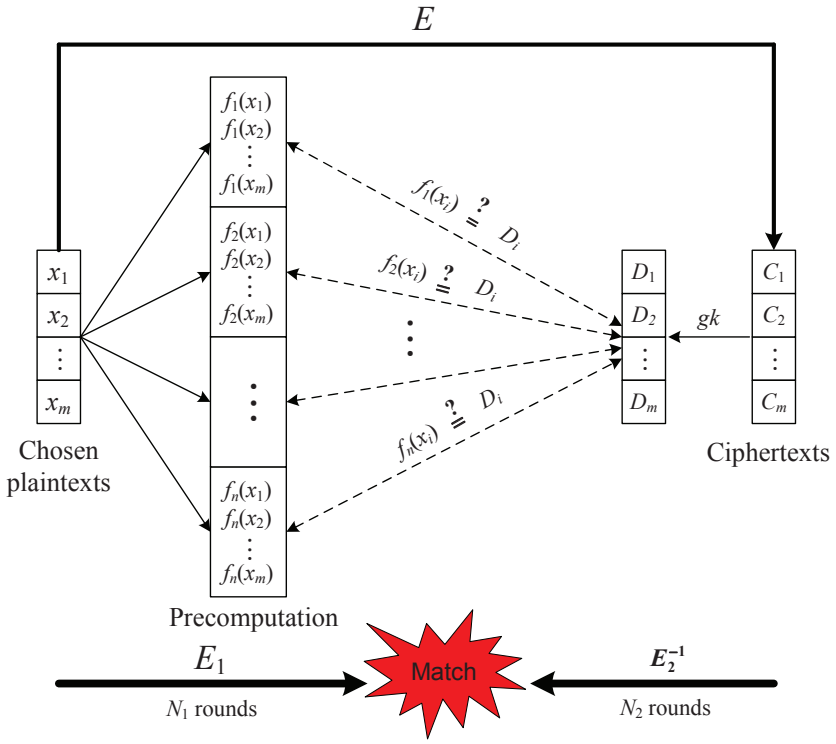

Fig. 1. The Meet-in-the-Middle Attack

The attack procedures are described in Fig.1:

Firstly, choose a set of m suitable plaintexts which are different at some fixed bits (denoted as $x_1, x_2, \ldots, x_m$), compute and store $f(x_i)$ for each possible f. This step is called the precomputation phase. Assume that there are n possible parameters for the function f, thus there are n possible functions f, denoted as $f_1, f_2, \ldots, f_n$.

Secondly, Encrypt the m plaintexts with N rounds and the ciphertexts are denoted as $C_1, C_2, \ldots, C_m$, then search certain subkey gk, do a partial N_2 rounds decryption and obtain $D_i = E_2^{-1}(C_i, gk)$, note that the position of D_i in the data state is the same as c_i, so they have the same length.

Thirdly, check whether $D_i = f_j(x_i)(1 \leq i \leq m)$ hold for some $f_j(1 \leq j \leq n)$, once an f_j is found so that $D_i = f_j(x_i)(1 \leq i \leq m)$, we call a match is found and the guessed subkey gk is mostly likely correct since the probability of having a match for a wrong key is approximately $n \times 2^{-k \times m}$, where k is the length of D_i, i.e. D_i is k-bit length. Then if m is big enough, all wrong keys can be excluded.

Note that in the precomputation phase, the number of the parameters in f can't be too large since the precomputation complexity would exceed the exhaustive search attack if n is too large. On the other hand, in the attack phases, sometimes we filtrate the wrong subkeys according to checking whether $f_j(x_i) \oplus f_j(x_{i'}) = D_i \oplus D_{i'}$ holds, because in this way we can reduce the precomputation complexity or guess less subkeys in the partial decryption phase. For the first case, we give an example: Assume that the function $f(x) = g(x) \oplus c$, where c is a parameter, then $f(x_i) \oplus f(x_{i'}) = g(x_i) \oplus g(x_{i'})$ and the parameter c can be ignored in the precomputation phase. For the second case, one will see it be used in our attacks on ARIA in Sec.4.

3 Description of ARIA

ARIA adopts a substitution-permutation network(SPN) and employs an involutional binary 16×16 matrix over $GF(2^8)$ in its diffusion layer. The substitution layer consists of sixteen 8×8-bit S-boxes based on the inversion in $GF(2^8)$. The 128-bit plaintext/ciphertext, as well as the input and output of the round function, are treated as 4×4 matrices with elements in $GF(2^8)$, depicted as follows:

x_0	x_4	x_8	x_{12}
x_1	x_5	x_9	x_{13}
x_2	x_6	x_{10}	x_{14}
x_3	x_7	x_{11}	x_{15}

The round function of ARIA firstly applies a *Round Key Addition*, then a *Substitution Layer* and at last a *Diffusion Layer* subsequently. An N-round ARIA iterates the round function $N - 1$ times; and in the last round, the diffusion layer is replaced by the *Round Key Addition*. The three operations are defined as follows:

Round Key Addition(RKA). The 128-bit round key is simply XORed to the state. The round keys are derived from the cipher key by means of the key schedule. We refer to ref.[1] for details.

Substitution Layer(SL). A non-linear byte substitution operates on each byte of the state independently which is implemented by two S-boxes S_1 and S_2. ARIA has two types of S-Box layers for odd and even rounds as shown in Fig.2. Type 1 is used in the odd rounds and type 2 is used in the even rounds.

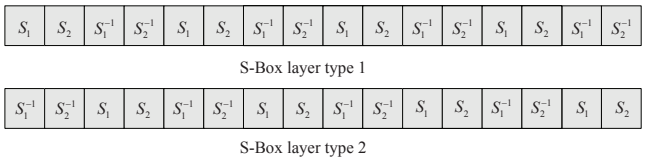

Fig. 2. The two types of S-Box layers

Diffusion Layer(DL). An involutional linear transformation $P : GF(2^8)^{16} \rightarrow GF(2^8)^{16}$ with branch number 8 is selected to improve the diffusion effect and increase efficiency in both hardware and software implementations [12]. The transformation P is given by

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

where

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

The Key Schedule of ARIA is omitted and we refer to ref. [1] for more details.

4 The Meet-in-the-Middle Attacks on ARIA

In this section, we first construct some 3/4-round distinguishing properties for the meet-in-the-middle attack on ARIA. Then we present some meet-in-the-middle attacks on the round-reduced ARIA based on the distinguishing properties.

4.1 3-Round Distinguishing Property of ARIA

In this subsection, we construct a 3-round distinguishing property of ARIA.

Definition 1. *A set* $\{a_i | a_i \in \mathbb{F}_{2^n}, 0 \leq i \leq 2^n - 1\}$ *is* active, *if for any* $0 \leq i < j \leq 2^n - 1$, $a_i \neq a_j$.

Definition 2. *A set* $\{a_i | a_i \in \mathbb{F}_{2^n}, 0 \leq i \leq 2^n - 1\}$ *is* passive, *if for any* $0 < i \leq 2^n - 1$, $a_i = a_0$.

In the following paper, C always denote some constant value but not necessarily equal to each other at different positions.

Let the input of ARIA be $B = (B_0, B_1, \ldots, B_{15})$, the i-th round key be $k_i = (k_{i,0}, k_{i,1}, \ldots, k_{i,15})$, and the outputs of S-Box layer and P layer of the i-th round be $Z_i = (Z_{i,0}, Z_{i,1}, \ldots, Z_{i,15})$ and $Y_i = (Y_{i,0}, Y_{i,1}, \ldots, Y_{i,15})$, respectively.

Consider the evolution of the plaintext over 3 inner rounds of ARIA, take a set of 256 plaintexts so that B_0 is an active byte and all the other bytes are passive, thus B_0 takes all values of $\mathbb{F}_{2^8}$ and B_is are constants where $1 \leq i \leq 15$. Let the input be

$$B = \begin{pmatrix} x & C & C & C \\ C & C & C & C \\ C & C & C & C \\ C & C & C & C \end{pmatrix},$$

and

$$y = S_1(x \oplus k_{1,0}), \tag{4}$$

then according to the definition of ARIA, the output of the first round is

$$Y_1 = \begin{pmatrix} C & y \oplus a_4 & y \oplus a_8 & C \\ C & C & y \oplus a_9 & y \oplus a_{13} \\ C & y \oplus a_6 & C & y \oplus a_{14} \\ y \oplus a_3 & C & C & C \end{pmatrix},$$

where a_is ($i = 3, 4, 6, 8, 9, 13, 14$) are some fixed values that depend on the passive bytes and subkey values.

Let $b_i = a_i \oplus k_{2,i}$, then

$$Z_2 = \begin{pmatrix} C & S_1^{-1}(y \oplus b_4) & S_1^{-1}(y \oplus b_8) & C \\ C & C & S_2^{-1}(y \oplus b_9) & S_2^{-1}(y \oplus b_{13}) \\ C & S_1(y \oplus b_6) & C & S_1(y \oplus b_{14}) \\ S_2(y \oplus b_3) & C & C & C \end{pmatrix}, \tag{5}$$

we denote Z_2 as

$$Z_2 \triangleq \begin{pmatrix} C & z_4 & z_8 & C \\ C & C & z_9 & z_{13} \\ C & z_6 & C & z_{14} \\ z_3 & C & C & C \end{pmatrix}, \tag{6}$$

and define

$$z(i,j,k,\ldots) = z_i \oplus z_j \oplus z_k \oplus \cdots, \tag{7}$$

thus

$$Y_2 = \begin{pmatrix} z(3,4,6,8,9,13,14) & z(8,14)\oplus c_4 & z(4,13)\oplus c_8 & z(6,9)\oplus c_{12} \\ z(8,9)\oplus c_1 & z(3,4,9,14)\oplus c_5 & z(6,14)\oplus c_9 & z(3,6,8,13)\oplus c_{13} \\ z(4,6)\oplus c_2 & z(9,13)\oplus c_6 & z(3,6,8,13)\oplus c_{10} & z(3,4,9,14)\oplus c_{14} \\ z(13,14)\oplus c_3 & z(3,6,8,13)\oplus c_7 & z(3,4,9,14)\oplus c_{11} & z(4,8)\oplus c_{15} \end{pmatrix},$$

where c_is for $1 \le i \le 15$ are some fixed values.

Let $d_i = c_i \oplus k_{3,i}$, then $Z_3 =$

$$\begin{pmatrix} S_1(z(3,4,6,8,9,13,14)\oplus k_{3,0}) & S_1(z(8,14)\oplus d_4) & S_1(z(4,13)\oplus d_8) & S_1(z(6,9)\oplus d_{12}) \\ S_2(z(8,9)\oplus d_1) & S_2(z(3,4,9,14)\oplus d_5) & S_2(z(6,14)\oplus d_9) & S_2(z(3,6,8,13)\oplus d_{13}) \\ S_1^{-1}(z(4,6)\oplus d_2) & S_1^{-1}(z(9,13)\oplus d_6) & S_1^{-1}(z(3,6,8,13)\oplus d_{10}) & S_1^{-1}(z(3,4,9,14)\oplus d_{14}) \\ S_2^{-1}(z(13,14)\oplus d_3) & S_2^{-1}(z(3,6,8,13)\oplus d_7) & S_2^{-1}(z(3,4,9,14)\oplus d_{11}) & S_2^{-1}(z(4,8)\oplus d_{15}) \end{pmatrix}.$$

We can summarize the above observations with the following theorem:

Theorem 1. (3-Round Distinguishing Property of ARIA) *Let the input of ARIA be $B = (B_0, B_1, \ldots, B_{15})$, the i-th round key be $k_i = (k_{i,0}, k_{i,1}, \ldots, k_{i,15})$, and the outputs of S-Box layer and P layer of the i-th round be $Z_i = (Z_{i,0}, Z_{i,1}, \ldots, Z_{i,15})$ and $Y_i = (Y_{i,0}, Y_{i,1}, \ldots, Y_{i,15})$, respectively. If B_0 takes all values of $\mathbb{F}_{2^8}$ and B_is are constants where $1 \le i \le 15$. Then, the function which maps B_0 to $Y_{3,0}$ is entirely determined by 15 fixed 1-byte parameters.*

Proof. From the above observations, we have

$$\begin{aligned} Y_{3,0} = {} & S_2^{-1}(z(13,14)\oplus d_3) \oplus S_1(z(8,14)\oplus d_4) \oplus S_1^{-1}(z(9,13)\oplus d_6) \oplus S_1(z(4,13)\oplus d_8) \\ & \oplus S_2(z(6,14)\oplus d_9) \oplus S_2(z(3,6,8,13)\oplus d_{13}) \oplus S_1^{-1}(z(3,4,9,14)\oplus d_{14}), \end{aligned} \tag{8}$$

and $z(i,j,k,\ldots)$ is the function of the variable x with the fixed 1-byte parameters $(k_{1,0}, b_i, b_j, b_k, \ldots)$. Therefore, the 15 fixed values

$$(k_{1,0}, b_3, b_4, b_6, b_8, b_9, b_{13}, b_{14}, d_3, d_4, d_6, d_8, d_9, d_{13}, d_{14}) \tag{9}$$

completely specify the mapping B_0 to $Y_{3,0}$. □

15 bytes is less to search exhaustively in an attack on all visions of ARIA, so this distinguishing property can be used to attack ARIA-128/192/256. Moreover, according to the encryption algorithm of ARIA, the distinguishing property shown in Theorem 1 can be generalized: The functions which map B_0 to $Y_{3,i}$ for $1 \le i \le 15$ all are entirely determined by 15 fixed 1-byte parameters, respectively. Similarly, any other B_i can be taken as the active byte instead of B_0.

4.2 4-Round Distinguishing property of ARIA

In this subsection, we extend the above 3-round distinguishing property of ARIA to 4-round one.

Theorem 2. (4-Round Distinguishing property of ARIA) *Let the input of ARIA be* $B = (B_0, B_1, \ldots, B_{15})$*, the i-th round key be* $k_i = (k_{i,0}, k_{i,1}, \ldots, k_{i,15})$*, and the outputs of S layer and P layer of the i-th round be* $Z_i = (Z_{i,0}, Z_{i,1}, \ldots, Z_{i,15})$ *and* $Y_i = (Y_{i,0}, Y_{i,1}, \ldots, Y_{i,15})$*, respectively. If* B_0 *takes all values of* $\mathbb{F}_{2^8}$ *and* B_i*s are constants where* $1 \leq i \leq 15$*. Then, the function which maps* B_0 *to* $Y_{4,1}$ *is entirely determined by 31 fixed 1-byte parameters.*

Proof. According to the expression of Z_3 in Sec.4.1, we have

$$\begin{cases} Y_{3,2} = S_2(z(8,9)\oplus d_1)\oplus S_1(z(8,14)\oplus d_4)\oplus S_1^{-1}(z(9,13)\oplus d_6)\oplus S_1^{-1}(z(3,6,8,13)\oplus d_{10}) \\ \quad \oplus S_2^{-1}(z(3,4,9,14)\oplus d_{11})\oplus S_1(z(6,9)\oplus d_{12})\oplus S_2^{-1}(z(4,8)\oplus d_{15}), \\ Y_{3,5} = S_2(z(8,9)\oplus d_1)\oplus S_2^{-1}(z(13,14)\oplus d_3)\oplus S_1(z(8,14)\oplus d_4)\oplus S_2(z(6,14)\oplus d_9) \\ \quad \oplus S_1^{-1}(z(3,6,8,13)\oplus d_{10})\oplus S_1^{-1}(z(3,4,9,14)\oplus d_{14})\oplus S_2^{-1}(z(4,8)\oplus d_{15}), \\ Y_{3,7} = S_2(z(8,9)\oplus d_1)\oplus S_2^{-1}(z(13,14)\oplus d_3)\oplus S_1^{-1}(z(9,13)\oplus d_6)\oplus S_1(z(4,13)\oplus d_8) \\ \quad \oplus S_2^{-1}(z(3,4,9,14)\oplus d_{11})\oplus S_1(z(6,9)\oplus d_{12})\oplus S_2(z(3,6,8,13)\oplus d_{13}), \\ Y_{3,8} = S_1(z(3,4,6,8,9,13,14)\oplus k_{3,0})\oplus S_2(z(8,9)\oplus d_1)\oplus S_1(z(8,14)\oplus d_4)\oplus S_2^{-1}(z(3,6,8, \\ \quad 13)\oplus d_7)\oplus S_1^{-1}(z(3,6,8,13)\oplus d_{10})\oplus S_2(z(3,6,8,13)\oplus d_{13})\oplus S_2^{-1}(z(4,8)\oplus d_{15}), \\ Y_{3,9} = S_1(z(3,4,6,8,9,13,14)\oplus k_{3,0})\oplus S_2(z(8,9)\oplus d_1)\oplus S_2(z(3,4,9,14)\oplus d_5)\oplus S_1^{-1}(z(9, \\ \quad 13)\oplus d_6)\oplus S_2^{-1}(z(3,4,9,14)\oplus d_{11})\oplus S_1(z(6,9)\oplus d_{12})\oplus S_1^{-1}(z(3,4,9,14)\oplus d_{14}), \\ Y_{3,12} = S_2(z(8,9)\oplus d_1)\oplus S_1^{-1}(z(4,6)\oplus d_2)\oplus S_1^{-1}(z(9,13)\oplus d_6)\oplus S_2^{-1}(z(3,6,8,13)\oplus d_7) \\ \quad \oplus S_2(z(6,14)\oplus d_9)\oplus S_2^{-1}(z(3,4,9,14)\oplus d_{11})\oplus S_1(z(6,9)\oplus d_{12}), \\ Y_{3,15} = S_2(z(8,9)\oplus d_1)\oplus S_1^{-1}(z(4,6)\oplus d_2)\oplus S_1(z(8,14)\oplus d_4)\oplus S_2(z(3,4,9,14)\oplus d_5) \\ \quad \oplus S_1(z(4,13)\oplus d_8)\oplus S_1^{-1}(z(3,6,8,13)\oplus d_{10})\oplus S_2^{-1}(z(4,8)\oplus d_{15}). \end{cases} \tag{10}$$

Thus

$$Y_{4,1} = S_1(Y_{3,2}\oplus k_{4,2})\oplus S_2^{-1}(Y_{3,5}\oplus k_{4,5})\oplus S_2(Y_{3,7}\oplus k_{4,7})\oplus S_1^{-1}(Y_{3,8}\oplus k_{4,8})\oplus S_2^{-1}(Y_{3,9}\oplus k_{4,9})\oplus S_1^{-1}(Y_{3,12}\oplus k_{4,12})\oplus S_2(Y_{3,15}\oplus k_{4,15}). \tag{11}$$

It's clearly that the 31 fixed 1-byte values

$$(k_{1,0}, b_3, b_4, b_6, b_8, b_9, b_{13}, b_{14}, k_{3,0}, d_1, \ldots, d_{15}, k_{4,2}, k_{4,5}, k_{4,7}, k_{4,8}, k_{4,9}, k_{4,12}, k_{4,15}) \tag{12}$$

are sufficient to express the function $B_0 \to Y_{4,1}$. □

31 bytes may be too much to search exhaustively in an attack on ARIA-128/192, but the distinguishing property can be used to attack ARIA-256. Similarly, the distinguishing property can be generalized: The functions which map B_0 to $Y_{4,i}$ for $0 \leq i \leq 15$ all are entirely determined by 31 fixed 1-byte parameters, respectively. Also, any other B_i can be taken as the active byte instead of B_0.

4.3 Attack on 5/6-Round ARIA

In this subsection, we describe the meet-in-the-middle attack on 6-round ARIA based on the 3-round distinguishing property detailedly and only present the analysis of the complexity for the 5-round attack.

The main ideas in the meet-in-the-middle attack are: We first precompute all possible $B_0 \rightarrow Y_{3,0}$ mappings according to Theorem 1. Then we choose and encrypt a suitable plaintext set and search certain key bytes, do a partial decryption on the ciphertext set, and compare the values obtained by this decryption to the values in the precomputed set. When a match is found, the key value tried is most likely the right key value.

Let the plaintext and ciphertext of ARIA be $P = (P_0, P_1, \ldots, P_{15})$ and $C = (C_0, C_1, \ldots, C_{15})$, respectively; the round key, the outputs of S-Box layer and P layer of the i-th round be $k_i = (k_{i,0}, k_{i,1}, \ldots, k_{i,15})$, $Z_i = (Z_{i,0}, Z_{i,1}, \ldots, Z_{i,15})$ and $Y_i = (Y_{i,0}, Y_{i,1}, \ldots, Y_{i,15})$, respectively.

In the following, we describe a meet-in-the-middle attack on 6-round ARIA.The attack is based on the 3-round distinguishing property in Theorem 1 with additional one round at the beginning and two rounds at the end as shown in Fig.3. Here we denote the mapping $B_0 \rightarrow Y_{3,0}$ as $x \rightarrow f(x)$.

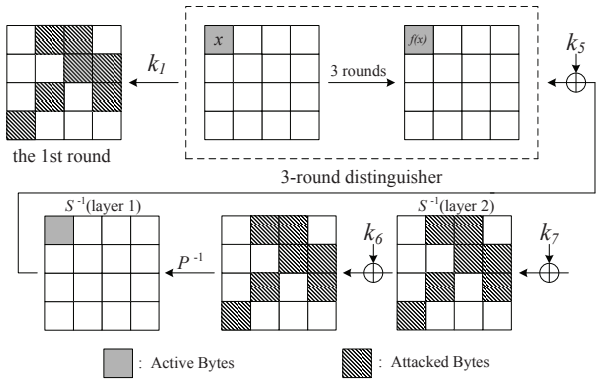

Fig. 3. Attack on 6-Round ARIA

The attack procedures are as follows:

Step 1 For each of the $2^{15\times 8}$ possible values of the parameters in (9), calculate the function $f : B_0 \rightarrow Y_{3,0}$, according to equations (4-8). For each f, compute and store

$$\Delta Y_{3,0}^{(i)} = f(i) \oplus f(0)$$

for $1 \leq i \leq 31$. In the following steps, we use $Y_{1,0}$ instead of B_0 and $Y_{4,0}$ instead of $Y_{3,0}$, the property is also holds.

Step 2 Guess $k_{1,3}, k_{1,4}, k_{1,6}, k_{1,8}, k_{1,9}, k_{1,13}, k_{1,14}$, choose a set plaintexts of the form

$$P = \begin{pmatrix} C & S_1^{-1}(x) \oplus k_{1,4} & S_1^{-1}(x) \oplus k_{1,8} & C \\ C & C & S_2^{-1}(x) \oplus k_{1,9} & S_2^{-1}(x) \oplus k_{1,13} \\ C & S_1(x) \oplus k_{1,6} & C & S_1(x) \oplus k_{1,14} \\ S_2(x) \oplus k_{1,3} & C & C & C \end{pmatrix}, \tag{13}$$

where $0 \leq x \leq 31$ and all the other bytes are constants, denote the 32 plaintexts as $P^{(i)}$ for $x = i$, encrypt all the 32 plaintexts with 6 rounds of ARIA, the corresponding ciphertexts denoted as $C^{(i)}$.

Step 3 Guess $k_{7,3}, k_{7,4}, k_{7,6}, k_{7,8}, k_{7,9}, k_{7,13}, k_{7,14}, k_6^*$, where $k_6^* = k_{6,3} \oplus k_{6,4} \oplus k_{6,6} \oplus k_{6,8} \oplus k_{6,9} \oplus k_{6,13} \oplus k_{6,14}$. For each ciphertext $C^{(i)}$, compute

$$Z_{5,0}^{(i)'} = S_2^{-1}(C_3^{(i)} \oplus k_{7,3}) \oplus S_1(C_4^{(i)} \oplus k_{7,4}) \oplus S_1^{-1}(C_6^{(i)} \oplus k_{7,6}) \oplus S_1(C_8^{(i)} \oplus k_{7,8}) \oplus S_2(C_9^{(i)} \oplus k_{7,9}) \oplus S_2(C_{13}^{(i)} \oplus k_{7,13}) \oplus S_1^{-1}(C_{14}^{(i)} \oplus k_{7,14}) \oplus k_6^*,$$

thus $Y_{4,0}^{(i)'} = S_1^{-1}(Z_{5,0}^{(i)'}) \oplus k_{5,0}$, then compute

$$\Delta Y_{4,0}^{(i)'} = Y_{4,0}^{(i)'} \oplus Y_{4,0}^{(0)'} = S_1^{-1}(Z_{5,0}^{(i)'}) \oplus S_1^{-1}(Z_{5,0}^{(0)'}),$$

so we need not to guess $k_{5,0}$.

Step 4 For each f, check whether

$$\Delta Y_{4,0}^{(i)} = \Delta Y_{4,0}^{(i)'}$$

holds for $1 \leq i \leq 32$.

Now if $k_{1,3}, k_{1,4}, k_{1,6}, k_{1,8}, k_{1,9}, k_{1,13}, k_{1,14}$ are guessed correctly, the 32 plaintexts after the first round encryption must be the form of

$$Y_1 = \begin{pmatrix} x & C & C & C \\ C & C & C & C \\ C & C & C & C \\ C & C & C & C \end{pmatrix},$$

where $0 \leq x \leq 31$ and all the other bytes are constants. Moreover, if $k_{7,3}, k_{7,4}, k_{7,6}, k_{7,8}, k_{7,9}, k_{7,13}, k_{7,14}, k_6^*$ are guessed correctly also, the function $Y_{1,0} \rightarrow Y_{4,0}$ must match one of the functions obtained in the precomputation phase, thus there must be an f so that $\Delta Y_{4,0}^{(i)} = \Delta Y_{4,0}^{(i)'}$ holds for $1 \leq i \leq 31$. Once a match is found, the corresponding $k_{1,3}, k_{1,4}, k_{1,6}, k_{1,8}, k_{1,9}, k_{1,13}, k_{1,14}, k_{7,3}, k_{7,4}, k_{7,6}, k_{7,8}, k_{7,9}, k_{7,13}, k_{7,14}, k_6^*$ are correct keys by an overwhelming probability, since the probability of having a match for a wrong key is approximately $2^{8\times 15} \times 2^{-8\times 31} = 2^{-128}$ and the number of the total guessed subkeys is 2^{120}.

Analysis of the attack complexity: According to the form of chosen plainlexts (13), we know that the data complexity is $2^{(16-9)\times 8} = 2^{56}$ since there are 9 bytes of constants; There is a precomputation step which calculates 2^{120} possible values for 32 plaintexts, therefore the complexity of this step is $32 \times 2^{120} = 2^{125}$ evaluations of the function and one evaluation of the function is equivalent to one round encryption of ARIA, so the precomputation complexity is about $2^{125}/6 \approx 2^{122.5}$; In the key search phase, one partial decryption is equivalent to 1/2 round encryption of ARIA, we need to guess total 15 bytes of subkeys, so the time complexity is $32 \times 2^{8\times 15}/(2 \times 6) \approx 2^{121.5}$.

For attacking on 5-round ARIA, it is based on the above 3-round distinguishing property with additional two rounds at the end. The attack procedures

are similar to the 6-round attack. In the chosen plaintext phase, we need only to choose 25 plaintexts, since in the key search phase, we need only guess 8 bytes, then the probability of having a match for a wrong key is approximately $2^{8\times 15} \times 2^{-8\times(25-1)} = 2^{-72}$, thus all wrong keys can be excluded. From the above analysis, we know the data complexity is 25; The precomputation complexity is also $2^{122.5}$; In the key search phase, we need only to guess 8 bytes, so the time complexity is $25 \times 2^{8\times 8}/(2 \times 5) \approx 2^{65.4}$.

4.4 Attack on 8-Round ARIA-256

In this subsection, we describe a meet-in-the-middle attack on 8-round ARIA-256. The attack is based on the 4-round distinguishing property in Theorem 2 with additional one round at the beginning and three rounds at the end as shown in Fig.4.

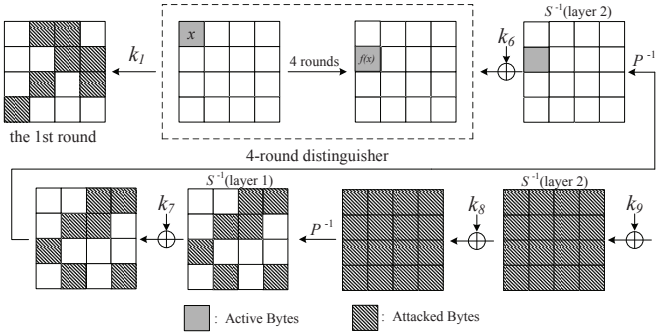

Fig. 4. Attack on 8-Round ARIA-256

The attack procedures are as follows:

Step 1 For each of the $2^{31\times 8}$ possible values of the parameters in (12), calculate the function $f : B_0 \rightarrow Y_{4,1}$, according to equations (4-7) and (10-11). For each f, compute and store

$$\Delta Y_{4,1}^{(i)} = f(i) \oplus f(0)$$

for $1 \leq i \leq 64$. In the following steps, we use $Y_{1,0}$ instead of B_0 and $Y_{5,1}$ instead of $Y_{4,1}$, the property is also holds.

Step 2 Guess $k_{1,3}, k_{1,4}, k_{1,6}, k_{1,8}, k_{1,9}, k_{1,13}, k_{1,14}$, choose a set plaintexts of the form

$$P = \begin{pmatrix} C & S_1^{-1}(x) \oplus k_{1,4} & S_1^{-1}(x) \oplus k_{1,8} & C \\ C & C & S_2^{-1}(x) \oplus k_{1,9} & S_2^{-1}(x) \oplus k_{1,13} \\ C & S_1(x) \oplus k_{1,6} & C & S_1(x) \oplus k_{1,14} \\ S_2(x) \oplus k_{1,3} & C & C & C \end{pmatrix},$$

where $0 \leq x \leq 64$ and all the other bytes are constants. Denote the 65 plaintexts as $P^{(i)}$ for $x = i$, the corresponding ciphertexts denoted as $C^{(i)}$. Encrypt all the 65 plaintexts with 8 rounds of ARIA.

Step 3 Guess all bytes of k_9 and $k^*_{8,2}, k^*_{8,5}, k^*_{8,7}, k^*_{8,8}, k^*_{8,9}, k^*_{8,12}, k^*_{8,15}, k^*_7$, where $k^*_8 = P^{-1}(k_8)$, $k^*_{8,i}$ is the i-th byte of k^*_8 and $k^*_7 = k_{7,2} \oplus k_{7,5} \oplus k_{7,7} \oplus k_{7,8} \oplus k_{7,9} \oplus k_{7,12} \oplus k_{7,15}$.

For each ciphertext $C^{(i)}$, let $D^{(i)} = S^{-1}(C^{(i)} \oplus k_9)$, then compute

$$\begin{cases} Z^{(i)'}_{7,2} = D^{(i)}_1 \oplus D^{(i)}_4 \oplus D^{(i)}_6 \oplus D^{(i)}_{10} \oplus D^{(i)}_{11} \oplus D^{(i)}_{12} \oplus D^{(i)}_{15} \oplus k^*_{8,2}, \\ Z^{(i)'}_{7,5} = D^{(i)}_1 \oplus D^{(i)}_3 \oplus D^{(i)}_4 \oplus D^{(i)}_9 \oplus D^{(i)}_{10} \oplus D^{(i)}_{14} \oplus D^{(i)}_{15} \oplus k^*_{8,5}, \\ Z^{(i)'}_{7,7} = D^{(i)}_1 \oplus D^{(i)}_3 \oplus D^{(i)}_6 \oplus D^{(i)}_8 \oplus D^{(i)}_{11} \oplus D^{(i)}_{12} \oplus D^{(i)}_{13} \oplus k^*_{8,7}, \\ Z^{(i)'}_{7,8} = D^{(i)}_0 \oplus D^{(i)}_1 \oplus D^{(i)}_4 \oplus D^{(i)}_7 \oplus D^{(i)}_{10} \oplus D^{(i)}_{13} \oplus D^{(i)}_{15} \oplus k^*_{8,8}, \\ Z^{(i)'}_{7,9} = D^{(i)}_0 \oplus D^{(i)}_1 \oplus D^{(i)}_5 \oplus D^{(i)}_6 \oplus D^{(i)}_{11} \oplus D^{(i)}_{12} \oplus D^{(i)}_{14} \oplus k^*_{8,9}, \\ Z^{(i)'}_{7,12} = D^{(i)}_1 \oplus D^{(i)}_2 \oplus D^{(i)}_6 \oplus D^{(i)}_7 \oplus D^{(i)}_9 \oplus D^{(i)}_{11} \oplus D^{(i)}_{12} \oplus k^*_{8,12}, \\ Z^{(i)'}_{7,15} = D^{(i)}_1 \oplus D^{(i)}_2 \oplus D^{(i)}_4 \oplus D^{(i)}_5 \oplus D^{(i)}_8 \oplus D^{(i)}_{10} \oplus D^{(i)}_{15}. \oplus k^*_{8,15} \end{cases}$$

thus

$$Z^{(i)'}_{6,1} = S_1(Z^{(i)'}_{7,2}) \oplus S^{-1}_2(Z^{(i)'}_{7,5}) \oplus S_2(Z^{(i)'}_{7,7}) \oplus S^{-1}_1(Z^{(i)'}_{7,8}) \oplus S^{-1}_2(Z^{(i)'}_{7,9}) \oplus S^{-1}_1(Z^{(i)'}_{7,12}) \oplus S_2(Z^{(i)'}_{7,15}) \oplus k^*_7,$$

and $Y^{(i)'}_{5,1} = S_2(Z^{(i)'}_{6,1}) \oplus k_{6,1}$, then compute

$$\Delta Y^{(i)'}_{5,1} = Y^{(i)'}_{5,1} \oplus Y^{(0)'}_{5,1} = S_2(Z^{(i)'}_{6,1}) \oplus S_2(Z^{(0)'}_{6,1}),$$

so we need not to guess $k_{6,1}$.

The rest steps are the same as the attack on 6-round ARIA. Note that in the attack on 8-round ARIA-256, there are 31 bytes of parameters in the 4-round distinguishing property and we need guess total 31 bytes of subkeys, so we let $0 \leq i \leq 64$ in the plaintexts to make sure the wrong subkeys all be discarded.

Analysis of the attack complexity: The data complexity is also 2^{56}; The precomputation complexity is $65 \times 2^{8\times 31} \approx 2^{254}$ evaluations of the function and one 8 rounds encryption of ARIA is equivalent to four evaluations of the function, so the precomputation complexity is about $2^{254}/4 \approx 2^{252}$; In the key search phase, one partial decryption is equivalent to 3/2 round encryption of ARIA, so the time complexity is $65 \times 2^{8\times 31} \times (3/2)/8 \approx 2^{251.6}$.

4.5 Attack on 7-Round ARIA-192

Based on the above 3/4-round distinguishing properties, we can't attack 7 rounds of ARIA-192. However, referring to the meet-in-the-middle attacks on AES in [10], we can improve the 4-round distinguishing property in Theorem 2 to get a new 4-round distinguishing property which can be used to attack 7 rounds of ARIA-192. In fact, it's a method that reduce the precomputation complexity at the cost of increasing the data and time complexities.

Theorem 3. (Improved 4-Round Distinguishing property of ARIA) *Let the input of ARIA be* $B = (B_0, B_1, \ldots, B_{15})$*, the i-th round key be* $k_i = (k_{i,0}, k_{i,1}, \ldots, k_{i,15})$*, and the outputs of S layer and P layer of the i-th round be* $Z_i = (Z_{i,0}, Z_{i,1}, \ldots, Z_{i,15})$ *and* $Y_i = (Y_{i,0}, Y_{i,1}, \ldots, Y_{i,15})$*, respectively. If* B_0 *takes all values of* $\mathbb{F}_{2^8}$ *and* B_i*s are constants where* $1 \leq i \leq 15$*. Then, the function which maps* B_0 *to* $Y_{4,1}$ *is entirely determined by 23 fixed 1-byte parameters with probability* 2^{-64}*.*

Proof. The parameters $(b_3, b_4, b_6, b_8, b_9, b_{13}, b_{14}, d_1, \ldots, d_{15})$ in the 4-round distinguishing property in Theorem 2 are entirely determined by the passive bytes when the key is fixed, and

$$Pr(b_3 = b_4 = b_6 = b_8 = b_9 = b_{13} = b_{14}, d_1 = d_2 = d_3) = 2^{-8\times 8} = 2^{-64}. \quad (14)$$

If we take $b = b_3 = b_4 = b_6 = b_8 = b_9 = b_{13} = b_{14}$ and $d = d_1 = d_2 = d_3$, then the function which maps B_0 to $Y_{4,1}$ is entirely determined by 23 fixed 1-byte parameters

$$(k_{1,0}, b, d, d_4, \ldots, d_{15}, k_{4,2}, k_{4,5}, k_{4,7}, k_{4,8}, k_{4,9}, k_{4,12}, k_{4,15}) \quad (15)$$

with probability 2^{-64}. □

Note that the chosen relations about parameters do not have any specific meaning. The number of equalities in (14) is chosen so that the complexity of the attack on 7-round ARIA-192 does not exceed the search exhaustively attack.

Based on the above 4-round distinguishing property, we can mount a successful attack on 7-round ARIA-192. The attack is based on the improved 4-round distinguishing property with additional one round at the beginning and two rounds at the end. The attack procedures are just similar to the attack on 6-round ARIA in Sec.4.3, here we omit the attack details and only analyze the attack complexity.

Since the improved 4-round distinguishing property holds with probability 2^{-64}, then in Step 2 of the attack in Sec.4.3, we choose 2^{64} sets of plaintexts of the form

$$P = \begin{pmatrix} C & S_1^{-1}(x) \oplus k_{1,4} & S_1^{-1}(x) \oplus k_{1,8} & C \\ C & C & S_2^{-1}(x) \oplus k_{1,9} & S_2^{-1}(x) \oplus k_{1,13} \\ C & S_1(x) \oplus k_{1,6} & C & S_1(x) \oplus k_{1,14} \\ S_2(x) \oplus k_{1,3} & C & C & C \end{pmatrix},$$

and we expect that the event

$$b_3 = b_4 = b_6 = b_8 = b_9 = b_{13} = b_{14}, d_1 = d_2 = d_3$$

occurs. Then the data complexity is $2^{64} \times 2^{56} = 2^{120}$; The precomputation complexity is $32 \times 2^{23\times 8}$ evaluations of the function and one evaluations of the function is equivalent to 3/2 round encryption of ARIA, so the precomputation complexity is $32 \times 2^{23\times 8} \times (3/2)/7 \approx 2^{187}$; In the key search phase, we need to guess 15 bytes also, and one partial decryption is equivalent to 1/2 round encryption of ARIA, so the time complexity is $32 \times 2^{64} \times 2^{8\times 15}/(2 \times 7) \approx 2^{185.3}$.

5 Experiments of Meet-in-the-Middle Attack on 3-round ARIA

For validating the correctness of the above meet-in-the-middle attacks on ARIA, we do some experiments on 3-round ARIA.

From Section 4.1, we know that

$$\begin{aligned} Y_{2,1} &= z(8,9) \oplus c_1 = z_8 \oplus z_9 \oplus c_1 \\ &= S_1^{-1}(S_1(x \oplus k_{1,0}) \oplus b_8) \oplus S_2^{-1}(S_1(x \oplus k_{1,0}) \oplus b_9) \oplus c_1, \end{aligned}$$

where $k_{1,0}, b_8, b_9$ and c_1 are 4 fixed 1-byte values. This is a 2-round distinguishing property of ARIA. The meet-in-the-middle attack on 3-round ARIA is based on the 2-round distinguishing property with additional one round at the end. In the precomputation phase, we compute and store $f(x_i) \oplus f(x_{i'})$, then the constant c_1 can be ignored.

Table 1. Experimental Results of Meet-in-the-Middle Attack on 3-round ARIA

Number of Chosen Plaintexts	Times of Success	Successful probability
5	493	49.3%
6	996	99.6%
7	1000	100%

The attack procedures are just similar to the above attacks, we only give the complexity analysis: The data complexity is only 6, the precomputation complexity is about 2^{24} since $f(x_i) \oplus f(x_{i'})$ is determined by only 3 fixed 1-byte values. In the key search phase, only one byte should be guessed, so the time complexity is about 6×2^8. Assume that the number of chosen plaintexts is n, then the probability of having a match for a wrong key is approximately $2^{8\times 3} \times 2^{-8\times(n-1)}$ and the number of the total guessed subkeys is 2^8, so the probability of the correct subkey can be determined uniquely is $1/(1+\lfloor 2^8 \times 2^{8\times 3} \times 2^{-8\times(n-1)} \rfloor)$ in theory. The successful probabilities are 50%, 100%, 100% in theory for $n = 5, 6, 7$, respectively. We have done 1000 times Experiments for choosing 5,6 and 7 plaintexts, respectively. Table 1 lists our experimental results. From which one can find that the successful probabilities are very closed to the theoretic analysis.

6 Conclusion

In this paper, we firstly construct some distinguishing properties of reduced round ARIA. These properties are based on the following observation: If one chooses a set of plaintexts, where one byte is active and all the other bytes are constants, after encrypting these plaintexts with 3 or 4 rounds of ARIA, all bytes of the output of 3rd or 4th round are determined by the initial active byte

and 15 or 31 fixed 1-byte constants. We then use these distinguishing properties to apply the meet-in-the-middle attack on 5/6/7/8 rounds of ARIA. All of these attacks have a huge precomputation and memory complexity, however, the precomputation only needs to compute one time.

Table 2 lists our works together with some known cryptanalytic results on ARIA, where Pre denotes the precomputation complexity. From table 2, one can find that the 5-round attack presented in this paper has the lowest data complexity and time complexity and the 6-round attack has the lowest data complexity comparing to the known results.

Table 2. Comparison of Attacks on ARIA

Attack	Rounds	Data	Time	Pre	Source
Impossible Differential	5	$2^{71.3}$	$2^{71.6}$	-	[5]
Boomerang Attack	5	2^{57}	$2^{115.5}$	-	[6]
Integral Attack	5	$2^{27.5}$	$2^{76.7}$	-	[7]
Meet-in-the-Middle Attack	5	25	$2^{65.4}$	$2^{122.5}$	Sec.4.3
Impossible Differential	6	2^{121}	2^{112}	-	[4]
Impossible Differential	6	$2^{120.5}$	$2^{104.5}$	-	[5]
Impossible Differential	6	2^{113}	$2^{121.6}$	-	[5]
Boomerang Attack	6	2^{57}	$2^{171.2}$	-	[6]
Integral Attack	6	$2^{124.4}$	$2^{172.4}$	-	[7]
Meet-in-the-Middle Attack	6	2^{56}	$2^{121.5}$	$2^{122.5}$	Sec.4.3
Truncated Differential	7	2^{81}	2^{81}	-	[3]
Meet-in-the-Middle Attack	7	2^{120}	$2^{185.3}$	2^{187}	Sec.4.5
Meet-in-the-Middle Attack	8	2^{56}	$2^{251.6}$	2^{252}	Sec.4.4

Acknowledgement

The work in this paper is partially supported by the Natural Science Foundation of China(No: 60803156) and the open research fund of State Key Laboratory of Information Security(No: 01–07).

References

1. Daesung Kwon, Jaesung Kim, Sangwoo Park and Soo Hak Sung etc. New Block Cipher: ARIA. In J.I.Lim and D.H.Lee(Eds.), ICISC 2003, LNCS 2971, pp.432-445, Springer-Verlag 2004.
2. Joan Daemen and Vincent Rijmen. The Design of Rijndael: AES — The Advanced Encryption Standard (Information Security and Cryptography). Springer, 2002.
3. Alex Biryukov, Christophe De Canniere, Joseph Lano, Siddika Berna Ors and Bart Preneel. Security and Performance Analysis of Aria. Version 1.2. Jan 7, 2004.
4. Wenling Wu, Wentao Zhang and Dengguo Feng. Impossible differential cryptanalysis of Reduced-Round ARIA and Camellia. In Journal of Computer Science and Technology 22(3), pp. 449-456, Springer-Verlag 2007.

5. Ruilin Li, Bing Sun, Peng Zhang and Chao Li. New Impossible Differentials of ARIA. Cryptology ePrint Archive, Report 2008/227, 2008. http://eprint.iacr.org/.
6. Ewan Fleischmann, Michael Gorski and Stefan Lucks. Attacking Reduced Rounds of the ARIA Block Cipher. Cryptology ePrint Archive, Report 2009/334, 2009. http://eprint.iacr.org/.
7. Ping Li, Bing Sun and Chao Li. Integral Cryptanalysis of ARIA. In pre-proceeding of Inscrypt 2009.
8. Hüseyin Demirci and Ali Aydın Selçuk. A meet in the middle attack on 8-round AES. Fast Software Encryption 2008. LNCS 5086, pp. 116-126. Springer-Verlag, 2008.
9. Whitfield Diffie and Martin E. Hellman. Exhaustive cryptanalysis of the NBS Data Encryption Standard. Computer Maganize, June 1977.
10. Hüseyin Demirci, İhsan Taşkın, Mustafa Çoban and Adnan Baysal. Improved Meet-in-the-Middle Attacks on AES. INDOCRYPT 2009, LNCS 5922, pp. 144-156. Springer-Verlag,2009.
11. Gilbert Henri and Marine Minier. A collision attack on 7 rounds of Rijndael. The Third AES Candidate Conference (2000)
12. Bon Wook Koo, Hwan Seok Jang and Jung Hwan Song, Constructing and Cryptanalysis of a 16×16 Binary Matrix as a Diffusion Layer. In K. Chae and M. Yung (Eds.): WISA 2003, LNCS 2908, pp.489-503, Springer-Verlag 2004.