Some results on ACORN

Dibyendu Roy and Sourav Mukhopadhyay
Department of Mathematics,
Indian Institute of Technology Kharagpur,
Kharagpur-721302, India
{dibyendu.roy, sourav}@maths.iitkgp.ernet.in

Abstract:- In this paper we obtain a weakness in the design specification of ACORN, which is a competitor of CAESAR competition. We show that there exists a probabilistic linear relation between message bits and ciphertext bits, which holds with probability greater than $\frac{1}{2}$. This is the first paper which finds a probabilistic linear relation between message and corresponding ciphertext bits of ACRON, and which holds with probability greater than $\frac{1}{2}$. We also propose a new type of CPA attack on ACORN. By our attack method, it is possible to recover full initial state of the encryption phase of the cipher, and the attack has complexity $\approx 2^{40}$. After obtaining the initial state of the encryption phase, one can invert the associated data loading phase and key-IV initialization phase to recover the secret key bits.

Keywords:- CAESAR, ACORN, linear approximation, CPA.

1 Introduction

ACORN [7, 8] is an authenticated encryption stream cipher, which is a submitted article in CAESAR competition [1]. It is a stream cipher based authenticated encryption cipher. The cipher is based on LFSRs, nonlinear feedback function and one nonlinear output function. The state size of the cipher is 293 and the cipher uses 128 bit secret key and 128 bit initialization vector to initialize the state. After the initialization phase is over the cipher starts producing the ciphertext bits which is simple XOR of keystream bits and plaintext bits like normal stream cipher. Further, the cipher generates tag bits, which are required for authentication. The decryption and verification are similar to the encryption and tag generation process.

Recently, Salam et al. [5] observed state collisions in ACORN. They have obtained collision for different cases such as: for two different associated data with same key, initialization vector and same plaintext; for two pairs of different key, initialization vector, and associated data with same plaintext. They have also obtained a collision for two different plaintext pairs. The basic idea for finding a collision is to, consider two different states in such a way that after a certain number of clockings the two states differ only at some desired positions. And for two different plaintexts or associated data, the difference between the two states cancel out and they generate same states. For doing this work one need to solve some system of equations involving a certain number of variables. In the same year, Lafitte et al. [4] have proposed a SAT-based attack on ACORN.

In ISC 2015, Jiao et al. [3] have tried to find a linear relation between the state bits of the cipher

ACORN by guessing some state bits, but they were unable to produce the final linear relation. In our work, we obtain a linear relation between the message bits and ciphertext bits, which is independent of other parameters.

In this paper, we obtain a probabilistic linear relation between plaintext bits and ciphertext bits, which holds with a probability greater than $\frac{1}{2}$ and the relation is independent of secret key or initialization vector. To obtain this kind of relation we need to find some linear approximation of nonlinear functions which are used in ACORN. In the last part of this article, we also propose a new type of attack on ACORN, by using a new chosen plaintext attack model. By this attack technique, we are able to recover the full state of the cipher in the first round of encryption phase. After that it is possible to recover the secret key by inverting the key-IV initialization and associated data loading phase. This is the first work which breaks ACORN with very practical complexity.

The rest of the article is organized as follows: In Section 2 we discuss about the design specification of ACORN. The probabilistic linear relation between message and ciphertext bits has been derived in Section 3. Our new CPA attack in described in Section 4. Finally, the article is concluded in Section 5.

2 Design specification of ACORN

In this section, we discuss about the design specification of ACORN ([7, 8]). This authenticated cipher is based on 128 bit key and 128 initialization vector. The total state size of the cipher is 293. The cipher is based on 6 LFSRs of different lengths 61, 46, 47, 39, 37 and 59 and one additional register of 4 bits. In the first phase, the cipher will be initialized by the 128-bit secret key and 128-bit IV then the cipher goes through the associated data processing phase. The final steps are encryption phase and tag generation phase. The design specification of the cipher is given in the following figure 1.

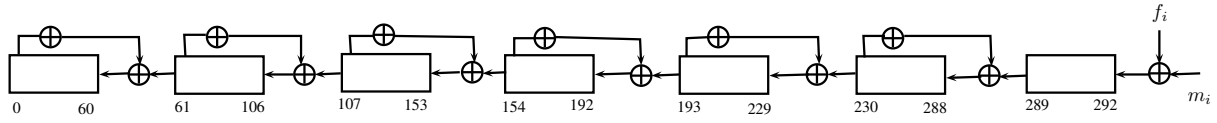

Figure 1: Design specification of ACORN

ACORN is based on 6 linear feedback functions, one nonlinear feedback function and one nonlinear output function. The i-th state bit at t-th clocking of the cipher is denoted by $s_{t,i}$ and the whole state at t-th clocking is denoted by s^t. In each clocking the feedback function of all 6 LFSRs are linear and the feedback function for the last bit is nonlinear. The expression of the nonlinear feedback function is given in the following expression,

$$\begin{aligned} f_t(s^t, a_t, b_t) =& 1 + s_{t,0} + s_{t,107} + s_{t,61} + s_{t,244}s_{t,23} + s_{t,23}s_{t,160} + s_{t,160}s_{t,244} + s_{t,230}s_{t,111} + s_{t,196}s_{t,111} \\ &+ s_{t,193}s_{t,111} + s_{t,230}s_{t,66} + s_{t,196}s_{t,66} + s_{t,193}s_{t,66} + a_t s_{t,196} + b_t(s_{t,12} + s_{t,154} + s_{t,111} \\ &+ s_{t,107} + s_{t,61}s_{t,193} + s_{t,61}s_{t,154} + s_{t,23}s_{t,193} + s_{t,23}s_{t,160} + s_{t,23}s_{t,154} + s_{t,0}s_{t,193} \\ &+ s_{t,0}s_{t,160} + s_{t,0}s_{t,154} + s_{t,193}s_{t,235} + s_{t,160}s_{t,235} + s_{t,154}s_{t,235} + s_{t,61}s_{t,235} + s_{t,23}s_{t,235} \\ &+ s_{t,0}s_{t,235}). \end{aligned}$$

In the expression of the nonlinear feedback function there are two parameters a_t and b_t which can take either 0 or 1 value. The values of these two parameters vary for different phases of the cipher.

So depending upon the values of a_t and b_t the expression of the nonlinear feedback function changes for different phases. The complete form of the state update function is given below,

State update function.

- $s_{t,289} = s_{t,289} + s_{t,235} + s_{t,230}$
- $s_{t,230} = s_{t,230} + s_{t,196} + s_{t,193}$
- $s_{t,193} = s_{t,193} + s_{t,160} + s_{t,154}$
- $s_{t,154} = s_{t,154} + s_{t,111} + s_{t,107}$
- $s_{t,107} = s_{t,107} + s_{t,66} + s_{t,61}$
- $s_{t,61} = s_{t,61} + s_{t,23} + s_{t,0}$
- $s_{t+1,j} = s_{t,j+1}$ for $j := 0$ to 291
- $s_{t+1,292} = f_t + m_t$

The cipher is based on one nonlinear output function and the expression is,

$$\begin{aligned} Y_t &= F_y(s_{t,12}, s_{t,61}, s_{t,154}, s_{t,193}, s_{t,235}) \\ &= s_{t,12} + s_{t,154} + s_{t,61}s_{t,193} + s_{t,193}s_{t,235} + s_{t,61}s_{t,235}. \end{aligned}$$

In the key-IV initialization phase and associated data processing phase, the cipher will be clocked for certain number of clockings without generating any output. In the encryption and tag generation phase the cipher produces ciphertext and tag as output.

Key-IV initialization phase. In this phase cipher will be initialized by the secret key and by the initialization vector. The secret key bits are denoted by k_i and the initialization vector bits are denoted by iv_i, where $i = 0, ..., 127$. In this phase the cipher will be clocked for 1792 clockings. The steps are given below,

- Initialize the whole state to 0; i.e., $s_i = 0$ for $i = 0, ..., 292$.
- $m_{-1792+i} = k_i$ and $m_{-1792+128+i} = iv_i$, for $i = 0$ to 127.
- $m_{-1792+256} = k_{i \mod 128} + 1$ for $i = 0$;
  $m_{-1792+256+i} = k_{i \mod 128}$ for $i = 1$ to 1535.
- Parameters $a_t = 1$ and $b_t = 1$ for all clockings.
- Update the state.

Associated data processing phase. After the key-IV initialization phase the cipher moves to the next phase, which is known as associated data processing phase. In this phase the cipher takes the associated data as input and update the state without producing any output bits. The general process is given below,

- $m_i = ad_i$ for $i = 0$ to $l - 1$, where l denotes the length of associated data and ad_i denotes the bits of associated data.
  $m_l = 1$.
  $m_{l+i} = 0$ for $i = 1$ to 255.

- $a_i = 1$ for $i = 0$ to $l + 127$;
  $a_i = 0$ for $i = l + 128$ to $l + 255$;
  $b_i = 1$ for $i = 0$ to $l + 255$.

- Update the state in each clocking for $i = 0$ to $m + 255$.

From the above discussion we can observe that if there is no associated data, still the cipher will be clocked for 256 clockings without generating any output bit. After that cipher moves to next phase.

Encryption. After the associated data processing phase of the cipher, it moves to the encryption phase. Here p_i denotes the plaintext bit and c_i denotes the corresponding ciphertext bit. The length of the plaintext is denoted by n. The general process of the encryption phase is given below,

- $m_i = p_i$ for $i = 0$ to $n - 1$;
  $m_n = 1$;
  $m_{n+i} = 0$ for $i = 1$ to 255.

- $a_i = 1$ for $i = 0$ to 383;
  $a_i = 0$ for $i = 384$ to 511;
  $b_i = 0$ for $i = 0$ to 511;

- Update the state in each clocking for $i = 0$ to 511;
  $c_i = p_i + Y_i$ for $i = 0$ to 511.

Tag generation phase. After the encryption phase the cipher moves to tag generation phase. In this phase the cipher generates tag corresponding to the plaintext, which are required for verification. The tag generation process is given below,

- $m_i = 0$ for $i = 0$ to 767.

- $a_i = 1$ for $i = 0$ to 767;
  $b_i = 1$ for $i = 0$ to 767.

- Update the state in each clocking for $i = 0$ to 767.

- The authentication tag bits are the last t keystream bits.

The decryption and verification phases are same as encryption and tag generation phase. The detailed description is given in the original article [7].

3 Existence of probabilistic linear relation between message and ciphertext bits only

In this section, we show that there exists a probabilistic relation between message and ciphertext bits of ACORN and the probability corresponding to this relation is greater than $\frac{1}{2}$. To find this linear relation between plaintext bits and ciphertext bits, we need to find some linear approximation of some nonlinear functions which are used in the design specification of ACORN. To find this relation we consider only the encryption phase of the cipher. We first find the linear approximation of the nonlinear feedback function. We consider only the expression when $a_i = 1$ in the encryption phase (as $a_i = 0$ means $s_{t,196}$ will be absent as linearly) and similar discussion can be done for $a_i = 0$.

Consider the expression of the nonlinear filter function with $a_t = 1$ and $b_t = 0$,

$$\begin{aligned} f_t(\cdot) =& 1 + s_{t,0} + s_{t,61} + s_{t,244}s_{t,23} + s_{t,160}(s_{t,23} + s_{t,244}) + (s_{t,230} + s_{t,196} + s_{t,193})(s_{t,66} + s_{t,111}) \\ &+ s_{t,107} + s_{t,196}. \end{aligned}$$

There are two nonlinear parts one is $A = s_{t,244}s_{t,23} + s_{t,160}(s_{t,23} + s_{t,244})$ and other one is $B = (s_{t,230} + s_{t,196} + s_{t,193})(s_{t,66} + s_{t,111})$.

Consider the following truth table 1,

$s_{t,244}$	$s_{t,23}$	$s_{t,160}$	A
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Table 1: Truth table related to the nonlinear filter function of ACORN.

From this truth table 1 we can easily observe that $Pr[A = s_{t,160}] = \frac{3}{4}$.

Now consider the expression of B. From the expression of B we can easily observe that $Pr[B = (s_{t,66} + s_{t,111})] = \frac{3}{4}$ as $B = 1$ iff $s_{t,230} + s_{t,196} + s_{t,193} = 1$ and $s_{t,66} + s_{t,111} = 1$ and 0 for all other cases.

From the above discussion we have,

$$\begin{aligned} &Pr[s_{t,244}s_{t,23} + s_{t,160}(s_{t,23} + s_{t,244}) = s_{t,160}] = \frac{3}{4} \\ &Pr[(s_{t,230} + s_{t,196} + s_{t,193})(s_{t,66} + s_{t,111}) = (s_{t,66} + s_{t,111})] = \frac{3}{4}. \end{aligned}$$

Now we will find the following probability,

$$\begin{aligned} &Pr[s_{t,244}s_{t,23} + s_{t,160}(s_{t,23} + s_{t,244}) + (s_{t,230} + s_{t,196} + s_{t,193})(s_{t,66} + s_{t,111}) = s_{t,160} + s_{t,66} + s_{t,111}] \\ &= Pr[A + B = s_{t,160} + s_{t,66} + s_{t,111}] \\ &= Pr[A = s_{t,160}]Pr[B = s_{t,66} + s_{t,111}] + Pr[A = 1 + s_{t,160}]Pr[B = 1 + s_{t,66} + s_{t,111}] \\ &= \frac{3}{4} \cdot \frac{3}{4} + \frac{1}{4} \cdot \frac{1}{4} \\ &= \frac{5}{8} \ = \ \frac{1}{2} + \frac{1}{8}. \end{aligned}$$

Hence,

$$Pr[f_t(\cdot) = 1 + s_{t,0} + s_{t,61} + s_{t,160} + s_{t,66} + s_{t,111} + s_{t,107} + s_{t,196}] = \frac{1}{2} + \frac{1}{8}. \tag{1}$$

Next we find linear approximation of the output function. The expression of the output function is,

$$\begin{aligned} Y^t &= F_y(s_{t,12}, s_{t,61}, s_{t,154}, s_{t,193}, s_{t,235}) \\ &= s_{t,12} + s_{t,154} + s_{t,61}s_{t,193} + s_{t,193}s_{t,235} + s_{t,61}s_{t,235}. \end{aligned}$$

Let $C = s_{t,61}s_{t,193} + s_{t,193}s_{t,235} + s_{t,61}s_{t,235}$. Consider the following truth table 2,

$s_{t,61}$	$s_{t,193}$	$s_{t,235}$	C	$1 + s_{t,61} + s_{t,193} + s_{t,235}$
0	0	0	0	1
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	0

Table 2: Truth table related to the output function of ACORN.

From this truth table 2, we can easily observe that $Pr[s_{t,61}s_{t,193} + s_{t,193}s_{t,235} + s_{t,61}s_{t,235} = 1 + s_{t,61} + s_{t,193} + s_{t,235}] = \frac{3}{4}$. Hence,

$$Pr[Y_t = s_{t,12} + s_{t,154} + 1 + s_{t,61} + s_{t,193} + s_{t,235}] = \frac{1}{2} + \frac{1}{4} \tag{2}$$

From the approximation 2, the linear approximation of the ciphertext bit will be,

$$c_i = p_i + s_{i,12} + s_{i,154} + 1 + s_{i,61} + s_{i,193} + s_{i,235}, \tag{3}$$

this equation will hold with probability $P = \left(\frac{1}{2} + \frac{1}{4}\right)$.

From the approximation 1, the linear approximation of the nonlinear feedback bit will be,

$$s_{i+1,292} = p_i + 1 + s_{i,0} + s_{i,61} + s_{i,160} + s_{i,66} + s_{i,111} + s_{i,107} + s_{i,196}, \tag{4}$$

this equation will hold with probability $Q = \left(\frac{1}{2} + \frac{1}{8}\right)$.

Now, by using the above two approximations we desire to find a probabilistic relation between message and ciphertext bits. To obtain this relation we implement the cipher along with these two approximations in SAGE software [6]. We first re-write the approximated output bit equation 3 in the following form,

$$c_i = p_i + X_i,$$

where $X_i = s_{i,12} + s_{i,154} + 1 + s_{i,61} + s_{i,193} + s_{i,235}$ (X_i denotes the terms involving only state bits). Now, we consider X_i's for 292 clockings. After implementing in SAGE software [6] we have

observed that these 292 equations (X_i) are linearly independent equations involving 292 initial state variables ($s_{0,j}$) only $s_{0,99}$ variable is absent. We have observed this linear independency by finding Gröbner bases [2] of this system of equations.

The expression of the feedback bit in the approximated form was,

$$\begin{aligned} s_{i+1,292} &= p_i + 1 + s_{i,0} + s_{i,61} + s_{i,160} + s_{i,66} + s_{i,111} + s_{i,107} + s_{i,196} \\ &= p_i + Y_i, \end{aligned}$$

where $Y_i = 1 + s_{i,0} + s_{i,61} + s_{i,160} + s_{i,66} + s_{i,111} + s_{i,107} + s_{i,196}$. To find a probabilistic relation between message bits and cipherext bits we need to sum the ciphertext bit equations in such a way that the state bits get canceled. To achieve this, we consider only X_i's and assume that feedback bits are only Y_i's but not the plaintext bits. Now, if we can find a relation $\sum_{i\in A} X_i = 0$, then in the relation $\sum_{i\in A} c_i$ all the state bits will get cancel only some plaintext, ciphertext bits will remain.

To obtain this type of relation we try to obtain one $X_j,\ j \notin \{0, ..., 291\}$, which can be written as linear combination of $\{X_0, ..., X_{291}\}$ by considering Y_i's as feedback bits. To find this we start from X_{292} and check whether it belongs to the ideal generated by polynomials $X_0, ..., X_{291}$. We do all these computations in SAGE software [6] and it has been observed that X_{325} belongs to the ideal generated by the polynomials $X_0, ..., X_{291}$, i.e., X_{325} can be expressed as linear combination of $X_0, ..., X_{291}$. From this result we can say that $\exists\ l_j \in \{0, 1\}$ such that $X_{325} = \sum_{j=0}^{291} l_j X_j$.

Now, our next aim is to find those $X_i \in \{X_0, ..., X_{291}\}$ which contributes in the sum $\sum_{j=0}^{291} l_j X_j$, i.e., we are interested to find the l_j's those are 1. To find this we have followed basis element replacement procedure. We choose one $X_k \in \{X_0, ..., X_{291}\}$ and check whether the equation X_k belongs to the ideal generated by the set of equations $\{X_0, ..., X_{k-1}, X_{325}, X_{k+1}, ..., X_{291}\}$ or not, if it belongs then the corresponding l_k is 1 otherwise l_k is 0, i.e., we choose any one $X_k \in \{X_0, ..., X_{291}\}$ and replace the X_k by X_{325} from the set $\{X_0, ..., X_{291}\}$ and check whether X_k belongs to the ideal generated by the new updated set or not, if it belongs then the corresponding coefficient l_k is 1 otherwise 0. By following this process we can find the l_i's those are 1. By using the SAGE software [6] we have observed that $l_i = 1$ for $i \in B = \{0,\ 3,\ 6,\ 12,\ 14,\ 15,\ 20,\ 21,\ 23,\ 24,\ 26,\ 27,\ 29,\ 33,\ 35,\ 36,\ 37,\ 38,\ 39,\ 41,\ 42,\ 44,\ 45,\ 46,\ 47,\ 50,\ 51,\ 57,\ 65,\ 69,\ 71,\ 74,\ 75,\ 76,\ 77,\ 78,\ 80,\ 81,\ 84,\ 88,\ 89,\ 93,\ 97,\ 101,\ 104,\ 105,\ 107,\ 109,\ 110,\ 113,\ 117,\ 118,\ 119,\ 122,\ 126,\ 127,\ 128,\ 130,\ 132,\ 133,\ 136,\ 137,\ 138,\ 140,\ 141,\ 143,\ 144,\ 145,\ 148,\ 149,\ 153,\ 154,\ 159,\ 160,\ 162,\ 163,\ 164,\ 169,\ 171,\ 178,\ 181,\ 183,\ 184,\ 185,\ 187,\ 188,\ 190,\ 191,\ 192,\ 197,\ 198,\ 199,\ 200,\ 202,\ 204,\ 206,\ 211,\ 215,\ 217,\ 221,\ 222,\ 223,\ 228,\ 229,\ 230,\ 231,\ 234,\ 238,\ 239,\ 240,\ 245,\ 247,\ 248,\ 249,\ 250,\ 251,\ 252,\ 259,\ 264,\ 266,\ 271,\ 278,\ 280,\ 282,\ 285,\ 286,\ 287,\ 288,\ 291\}$. So, now we can write X_{325} by the linear combination of the equations $\{X_0, ..., X_{291}\}$ and the linear combination is $X_{325} = \sum_{j=0}^{291} l_j X_j$, where $l_j = 1,\ \forall j \in B$ and $l_j = 0,\ \forall j \notin B$.

From the above discussion we can observe that $X_{325} + \sum_{j=0}^{291} l_j X_j = 0$ (state bits get cancel), where $l_j = 1,\ \forall j \in B$ and $l_j = 0,\ \forall j \notin B$. Now we consider the sum $\sum_{j\in B_1} c_j$, where $B_1 = B \cup \{325\}$.

$$\sum_{j\in B_1} c_j = \sum_{j\in B_1} p_j + \sum_{j\in B_1} X_j. \tag{5}$$

Initially to find the relation between X_j's we did not consider the plaintext bits in feedback function, because we wanted to find one relation between X_j's such that in the final relation all the state bits vanishes. Due to the presence of plaintext bits in the feedback bit equation, there will be some plaintext bits in $\sum_{j\in B_1} X_j$, but there will not be any state bits (as they cancel out). We have observed that the final expression of the sum $\sum_{j\in B_1} p_j + \sum_{j\in B_1} X_j$ is,

$$\begin{aligned}\sum_{j\in B_1} p_j + \sum_{j\in B_1} X_j =& p_0 + p_1 + p_3 + p_7 + p_9 + p_{10} + p_{12} + p_{15} + p_{17} + p_{18} + p_{19} + p_{23} + p_{24} + p_{26} \\ &+ p_{27} + p_{30} + p_{32} + p_{34} + p_{35} + p_{38} + p_{40} + p_{42} + p_{43} + p_{44} + p_{45} + p_{46} \\ &+ p_{49} + p_{50} + p_{51} + p_{54} + p_{55} + p_{57} + p_{58} + p_{60} + p_{62} + p_{67} + p_{68} + p_{70} \\ &+ p_{72} + p_{73} + p_{77} + p_{78} + p_{79} + p_{81} + p_{82} + p_{88} + p_{90} + p_{92} + p_{93} + p_{97} \\ &+ p_{101} + p_{106} + p_{109} + p_{112} + p_{113} + p_{115} + p_{116} + p_{117} + p_{122} + p_{123} \\ &+ p_{124} + p_{125} + p_{127} + p_{131} + p_{132} + p_{135} + p_{136} + p_{138} + p_{140} + p_{142} \\ &+ p_{143} + p_{144} + p_{145} + p_{147} + p_{158} + p_{159} + p_{160} + p_{163} + p_{164} + p_{167} \\ &+ p_{169} + p_{170} + p_{171} + p_{173} + p_{176} + p_{178} + p_{179} + p_{180} + p_{181} + p_{182} \\ &+ p_{183} + p_{184} + p_{185} + p_{186} + p_{187} + p_{188} + p_{193} + p_{194} + p_{195} + p_{197} \\ &+ p_{198} + p_{199} + p_{200} + p_{204} + p_{206} + p_{207} + p_{211} + p_{215} + p_{217} + p_{222} \\ &+ p_{225} + p_{226} + p_{238} + p_{239} + p_{240} + p_{245} + p_{247} + p_{248} + p_{249} + p_{250} \\ &+ p_{251} + p_{252} + p_{259} + p_{264} + p_{266} + p_{268} + p_{271} + p_{278} + p_{280} + p_{282} \\ &+ p_{285} + p_{286} + p_{287} + p_{288} + p_{291} + p_{325}.\end{aligned} \tag{6}$$

We denote the index set corresponding to above all p_i's by A_1. Finally, we have one probabilistic relation between plaintext bits and ciphertext bits, and the relation is,

$$\sum_{j\in B_1} c_j = \sum_{j\in A_1} p_i. \tag{7}$$

Now we need to find the probability corresponding to this relation 7. To obtain the relation 7 we need to add the probabilistic ciphertext bits equations for $|B_1|$ number of times, where $|B_1|$ denotes the cardinality of the set B_1 and the cardinality is 130. Also we have replaced many feedback bits by its probabilistic expression in terms of the initial state bits. Replacing one feedback bit from one expression means add the expression of the feedback bit with the considered expression. It has been observed that if we consider the sum $\sum_{i\in B_1} c_i$ then we need to replace the feedback bits for 106 times, i.e., we need to add the probabilistic feedback bit equation 4 with the original equation for 106 times for different clockings. Ciphertext bit expressions are independent to each other, as different ciphertext bit equations contain different plaintext bits. Also, the feedback bit equations are independent from the output bit equations. Now, by Piling up lemma the probability

corresponding to the relation 7 will be,

$$\begin{aligned}\mathbf{P} &= \frac{1}{2} + \left(2^{129} \times \left(\frac{1}{4}\right)^{130}\right) \times \left(2^{105} \times \left(\frac{1}{8}\right)^{106}\right) \\ &= \frac{1}{2} + \frac{1}{2^{344}}.\end{aligned}$$

Finally, we are able to obtain a probabilistic relation between plaintext bits and ciphertext bits, which is independent of secret key bits, and the relation holds with probability $\mathbf{P} = \left(\frac{1}{2} + \frac{1}{2^{344}}\right)$. The relation is,

$$F(p, c) \Rightarrow \sum_{j \in B_1} c_j = \sum_{j \in A_1} p_i. \tag{8}$$

From this discussion the following theorem follows,

Theorem 3.1. *There exists a linear probabilistic relation F between the message bits and the corresponding ciphertext bits only, which holds with probability $\boldsymbol{P} = \frac{1}{2} + \frac{1}{2^{344}}$.*

Note: We have seen that X_{325} was the first equation which belongs to the ideal generated by the equations $X_0, ..., X_{291}$. Now if we consider any X_j for $j > 325$ which belongs to the ideal generated by the same set of equations, then in the final relations the number of feedback bits will increase, as in the expression of X_j for $j > 325$ has more feedback bits than X_{325}. Due to this reason the bias corresponding to the final linear relation will decrease. Hence it can be said that this is the best linear relation, which exists between message and ciphertext bits of ACORN, which has highest bias. In ISC 2015, Jiao et al. [3] have tried to find a probabilistic linear relation between the state bits of ACORN, in this paper we have come with a probabilistic linear relation between message and ciphertext bits only.

4 New attack on ACORN

In this section, we discuss about our new attack method on ACORN. In the expression of the state update function, we can note that the expression of the last feedback bit is $s_{t+1,292} = f_t + m_t$, where f_t is a function involving the current state bits of the cipher. The value of the feedback bit is either 0 or 1. It may happen that for one key, IV and associated data pair, one can find one message M, such that the value of the last feedback bit will be 0 for all clockings, i.e., $s_{t+1,292} = 0$ for all clockings, this implies $f_t + m_t = 0 \Rightarrow m_t = f_t$ for all clockings starting from the first clocking. If this type of situation occurs then the last register of 4-bit will not affect the ciphertext after 4 clockings, because the state of the last register will be null after 4 clocking. Due to this reason the affect of the nonlinear feedback function will not be present in the state update function of the cipher, due to this the degree of the ciphertext bit equation will remain fix. So, for one key, IV and associated pair there exists one message M which will help us to construct very low degree keystream bit equations (constant degree), involving the state bits of the cipher. The expression of the message bits will be,

$$\begin{aligned}m_t &= 1 + s_{t,0} + s_{t,107} + s_{t,61} + s_{t,244}s_{t,23} + s_{t,23}s_{t,160} + s_{t,160}s_{t,244} + s_{t,230}s_{t,111} + s_{t,196}s_{t,111} \\ &\quad + s_{t,193}s_{t,111} + s_{t,230}s_{t,66} + s_{t,196}s_{t,66} + s_{t,193}s_{t,66} + a_t s_{t,196} \\ &= f_t(s^t, a_t),\end{aligned}$$

where $a_t = 1$ for $t = 0, ..., 383$ and $a_t = 0$ for $t = 384, ..., 511$. So the expression of the message bits will be,

$$\begin{aligned} m_t =& 1 + s_{t,0} + s_{t,107} + s_{t,61} + s_{t,244}s_{t,23} + s_{t,23}s_{t,160} + s_{t,160}s_{t,244} + s_{t,230}s_{t,111} + s_{t,196}s_{t,111} \\ &+ s_{t,193}s_{t,111} + s_{t,230}s_{t,66} + s_{t,196}s_{t,66} + s_{t,193}s_{t,66} + s_{t,196}, \end{aligned}$$

for $t = 0, ..., 383$, and

$$\begin{aligned} m_t =& 1 + s_{t,0} + s_{t,107} + s_{t,61} + s_{t,244}s_{t,23} + s_{t,23}s_{t,160} + s_{t,160}s_{t,244} + s_{t,230}s_{t,111} + s_{t,196}s_{t,111} \\ &+ s_{t,193}s_{t,111} + s_{t,230}s_{t,66} + s_{t,196}s_{t,66} + s_{t,193}s_{t,66}, \end{aligned}$$

for $t = 384, ..., 511$.

Now, if we substitute the expression of the message in terms of the state bits in the expression of the ciphertext bit equation, then the expression of the ciphertext bit equation will be,

$$\begin{aligned} c_t &= z_t + m_t \\ &= F_t + f_t \\ &= s_{t,12} + s_{t,154} + s_{t,61}s_{t,193} + s_{t,193}s_{t,235} + s_{t,61}s_{t,235} + 1 + s_{t,0} + s_{t,107} + s_{t,61} + s_{t,244}s_{t,23} \\ &+ s_{t,23}s_{t,160} + s_{t,160}s_{t,244} + s_{t,230}s_{t,111} + s_{t,196}s_{t,111} + s_{t,193}s_{t,111} + s_{t,230}s_{t,66} + s_{t,196}s_{t,66} \\ &+ s_{t,193}s_{t,66} + a_t s_{t,196}, \end{aligned} \tag{9}$$

where $a_t = 1$ for $t = 0, ..., 383$ and $a_t = 0$ for $t = 384, ..., 511$. As the last feedback bit will be 0 for all clocking corresponding to the message M, where m_t's are the message bits and $m_t = f_t$, then the degree of the ciphertext bit equations will remain same for all clocking and the degree will be 2. Now, if attacker knows the ciphertext bits corresponding to that desired plaintext bits m_t, then he can construct a system of equations involving the state bits of cipher, the form of the equation is given in equation 9. Now the question is that how the adversary will able to find the ciphertext corresponding to the desired message M.

To find the values of the ciphertext bits for all clockings corresponding to the desired message M, adversary will use the cipher ACORN as black box in the encryption phase. We assume that the cipher has gone through key-IV initialization phase and associated data loading phase corresponding to one key-IV associated data pair (K, IV, ad). Adversary will consider the following function,

$$\begin{aligned} f_t(s^t, a_t) =& 1 + s_{t,0} + s_{t,107} + s_{t,61} + s_{t,244}s_{t,23} + s_{t,23}s_{t,160} + s_{t,160}s_{t,244} + s_{t,230}s_{t,111} + s_{t,196}s_{t,111} \\ &+ s_{t,193}s_{t,111} + s_{t,230}s_{t,66} + s_{t,196}s_{t,66} + s_{t,193}s_{t,66} + a_t s_{t,196}, \end{aligned} \tag{10}$$

where $a_t = 1$ for $t = 0, ..., 383$ and $a_t = 0$ for $t = 384, ..., 511$ and s^t denotes the state of the cipher at t-th clocking. Then adversary asks the black box (challenger), in each clocking you apply this function on your current state and consider the output of the function as plaintext bit, then encrypt the plaintext bit and provide me the ciphertext bit c_t for all clocking. Then the black box (challenger) does the following computations, it performs this function $f_t(s^t, a_t)$ on current state of the cipher at t-th clocking and consider that bit as message bit $m_t = f_t(s^t, a_t)$ and encrypt that message bit and output ciphertext bits c_t for all clockings.

Now the adversary is having the ciphertext bits c_t corresponding to message bits m_t for all clockings. We can observe that this ciphertext $C = c_0c_1...$ is the ciphertext corresponding to the desired message $M = m_0m_1....$ By using the ciphertext bits c_t for all t, adversary can construct fixed

degree equations (of degree 2) for different clockings, these equations involve only the initial state bits of the cipher. The form of the equation is described in equation 9.

Finally, the adversary solves the system to get back the values of the initial state bits $s_{0,i}$, $i = 0, ..., 292$, of the encryption phase of the cipher. We can observe that the degree of the equation 9 is 2 and it remains same for all clockings, as this message M makes the nonlinear feedback to 0 for all clocking. Now if we linearize the system, we need maximum $T = \binom{293}{2} = 42778$ number of variables. After linearizing the system, we can solve the system by Gauss elimination method. As the modern days computers can do 64 bit operations in one CPU clocks, then we need $\frac{7T^{\log_2^7}}{64} \approx 2^{40}$ CPU clocks to solve the system. Hence, the complexity of our attack is very much lesser than the complexity of exhaustive search on the security parameters. For practical implementation one can use SAGE [6] to recover the initial state bits. Now after getting the initial state of the cipher in the encryption phase of the cipher, adversary will run the inverse algorithms of associated data loading phase and key-IV initialization phase to recover the secret key, which can be done very easily after knowing the initial state of the encryption phase of ACORN.

Hence, we can observe that an adversary can recover the initial state of encryption phase of ACORN in 2^{40} complexity, and further can recover the secret key by using inverse algorithms of associated data loading phase and key-IV loading phase, which can be done very easily. The complexity of the attack is less than the complexity of the exhaustive search on the security parameter of the cipher, which is 2^{128}.

5 Conclusion

In this paper, we have proved the existence of linear relation between message and ciphertext bits only. We have shown that there exists a linear relation between message and corresponding ciphertext bits, which holds with probability greater than $\frac{1}{2}$ even if the cipher uses a secret key & IV to initialize the state, and in the encryption phase the state bits are involved nonlinearly with the message. We can also observe that this is the best possible linear relation exists between the message and ciphertext bits of ACORN. We have also proposed a new attack on ACORN, by this attack method it is possible to recover the initial state of the encryption phase of the cipher in 2^{40} complexity. Further one can recover the secret key by inverting the associated data loading algorithm and key-IV initialization algorithm. Hence, it can be seen that we are able to break ACORN in 2^{40} complexity, which is lesser than the complexity of exhaustive search on the security parameter.

References

[1] CAESAR: Competition for authenticated encryption: Security, applicability, and robustness

[2] Buchberger, B.: Gröbner bases: An algorithmic method in polynomial ideal theory. Multidimensional systems theory pp. 184–232 (1985)

[3] Jiao, L., Zhang, B., Wang, M.: Two generic methods of analyzing stream ciphers. In: International Information Security Conference. pp. 379–396. Springer (2015)

[4] Lafitte, F., Lerman, L., Markowitch, O., Van Heule, D.: Sat-based cryptanalysis of acorn

[5] Salam, M.I., Wong, K.K.H., Bartlett, H., Simpson, L., Dawson, E., Pieprzyk, J.: Finding state collisions in the authenticated encryption stream cipher acorn. Tech. rep., Cryptology ePrint Archive, Report 2015/918, 2015. http://eprint.iacr.org/2015/918.pdf

[6] Stein, W., et al.: Sage: Open source mathematical software (2008)

[7] Wu, H.: ACORN: A lightweight authenticated cipher (v1). CAESAR First Round Submission, competitions. cr. yp. to/round1/acornv1. pdf (2014)

[8] Wu, H.: ACORN: A lightweight authenticated cipher (v2). CAESAR Second Round Submission, competitions. cr. yp. to/round2/acornv2. pdf (2015)