Cryptanalysis of Stream Cipher Grain Family *

Haina Zhang[1], and Xiaoyun Wang[1,2]

[1] Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan 250100, China
hnzhang.cn@gmail.com
[2] Center for Advanced Study, Tsinghua University, Beijing 100084, China
xiaoyunwang@mail.tsinghua.edu.cn

Abstract. Grain v1 is one of the 7 final candidates of ECRYPT eStream project, which involves in the 80-bit secret key. Grain-128 is a variant version with 128-bit secret key, and Grain v0 is the original version in the first evaluation phase. Firstly, we describe a distinguishing attack against the Grain family with weak Key-IVs. Utilizing the second Walsh spectra of the nonlinear functions, we show that there are $2^{64}/2^{64}/2^{96}$ weak Key-IVs among total $2^{144}/2^{144}/2^{224}$ Key-IVs, and to distinguish a weak Key-IV needs about $2^{12.6}/2^{44.2}/2^{86}$ keystream bits and $2^{15.8}/2^{47.5}/\ 2^{104.2}$ operations for Grain v0, Grain v1 and Grain-128 respectively. Secondly, we apply algebraic attacks to the Grain family with a weak Key-IV, and can recover the secret key in about 2 seconds and 150 keystream bits for Grain v0 and Grain v1, and reveal the key of Grain-128 with about 100 keystream bits and $2^{93.8}$ operations. Furthermore, we discuss the period of the keystream with a weak Key-IV for any Grain-like structure which can lead in self-sliding attack.

Key Words: ECRYPT, eStream, stream cipher, Grain, Walsh spectra, algebraic attack.

1 Introduction

In the past ten years, many cryptographers focus on the development of stream ciphers because of two arresting projects which call for stream cipher primitives. The first one is the NESSIE [14] project launched in 1999, and no available stream cipher candidate to be selected as the final winner among 8 proposals. In 2004, the European Network of Excellence in Cryptology (ECRYPT) started a new call for stream cipher proposals named eStream project [5], there are total 34 candidates were submitted in the first evaluation phase. After three evaluation phases, there are 7 candidates left, and turned into the portfolio at September 9 in 2008. Grain v1 [8] is one of the 7 final algorithms, and Grain-128 [8] is a variant version with 128-bit secret key, and Grain v0 [9] is its original proposal in the first phase.

* Supported by the National Natural Science Foundation of China (NSFC Grant No.90604036) and 973 Project (No.2007CB807902).

Grain family is oriented to hardware applications, maintains a very low hardware cost. The design is based on two shift registers, one is a linear feedback shift register (LFSR), and the other is a nonlinear feedback shift register (NFSR). The LFSR guarantees a minimum period for the keystream. The NFSR, together with a nonlinear filter provides the nonlinearity to the cipher. Both for Grain v0 and Grain v1, the secret key is 80 bits, and the IV is specified to be 64 bits. For Grain-128, the key and IV are selected as 128 and 96 bits respectively.

In 2005, Khazaei, Hassanzadeh and Kiaei presented a distinguishing attack on Grain v0 [11]. At FSE 2006, Berbain, Gilbert and Maximov showed two key recovery attacks against Grain v0 [3]. The proposed key recovery attacks exploit 16 linear approximations of the filter function. The first one requires 2^{55} operations, 2^{49} bits memory and 2^{51} keystream bits, and the second requires 2^{43} operations, 2^{42} bits memory and 2^{38} keystream bits. In response to the attacks, the designers improved the algorithm, and submitted the new version Grain v1 and its 128-bit key variant Grain-128 to the second evaluation phase.

Based on the slide resynchronization attack, ö. Küçük presented a related key attack on Grain v1. Then two research groups extended the attack in [10] and proposed related-key chosen IV attacks on Grain-v1 and Grain-128 [4, 12]. All the attacks on Grain-v1 and Grain-128 are related-key settings. The algebraic attacks on two algorithms were proposed in [1], however the total time complexity exceeds the exhaustive attack.

In this paper, we describe a distinguishing and key recovery attack against the weak Key-IVs of Grain family. Utilizing the second Walsh spectra of the nonlinear functions, we present that there are 2^{64}, 2^{64} and 2^{96} weak Key-Ivs for Grain v0, Grain v1 and Grain-128 respectively. For Grain v0, to distinguish the weak Key-IVs only need $2^{12.6}$ keystream bits, the time complexity is about $2^{15.8}$. For Grain v1, the distinguishing attack needs $2^{44.2}$ keystream bits, and the time complexity is about $2^{47.5}$. For Grain-128, 2^{86} keystream bits and $2^{104.2}$ operations are required. Secondly, we apply algebraic attacks against the weak Key-Ivs of the Grain family, and show that the weak Key-IVs of Grain v0 and Grain v1 can be broken in about 2 second with about 150 keystream bits. To break the weak Key-IVs of Grain-128, 100 keystream bits and $2^{93.8}$ operations are required. Furthermore, we discuss the periods of the weak Key-IVs which lead to generalized distinguishing attacks against the Grain-like structure.

This paper is organized as follows. We first describe three algorithms in the Grain family in Section 2. We discuss the existence of the weak Key-IVs in Section 3, and propose a distinguishing attack against the weak Key-IVs of Grain family respectively in Section 4. Section 5 presents the key recovery of the weak Key-IVs by algebraic attacks. In Section 6, we further discuss the distinguishing attacks with the periods of the NFSRs. Finally, we conclude the paper in Section 7.

2 A Brief Description of Grain Family

Grain is based upon three main building blocks: a k-bit linear feedback shift register (LFSR), a k-bit nonlinear feedback shift register (NFSR), and a nonlinear filtering function, where $k = 80$, or 128. Grain is initialized with the k-bit key K and the l-bit initialization value IV. The cipher output is an L-bit keystream sequence $(z_t)_{t=0,\cdots,L-1}$. The structure is illustrated in Fig.1. The content of the LFSR is denoted by s_i, s_{i+1}, $\cdots$, s_{i+k-1} and the content of the NSFR is denoted by b_i, b_{i+1}, $\cdots$, b_{i+k-1}.

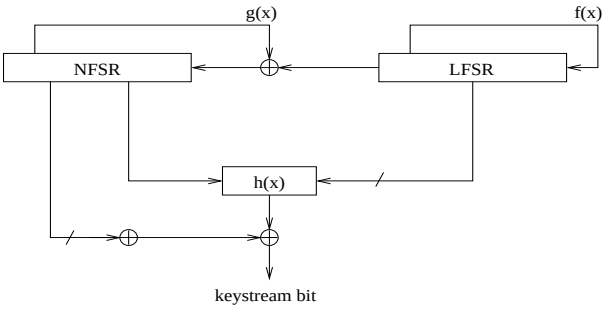

Fig. 1. The cipher body of Grain

2.1 The Functions of Grain Family

The Functions of Grain v0: For Grain v0, $k = 80$, $l = 64$. The feedback polynomial $f_0(x)$ of the LFSR is a primitive polynomial of degree 80. It is defined as

$$f_0(x) = 1 + x^{18} + x^{29} + x^{42} + x^{57} + x^{67} + x^{80}.$$

i.e.,

$$s_{t+80} = s_{t+62} + s_{t+51} + s_{t+38} + s_{t+23} + s_{t+13} + s_t.$$

The feedback polynomial $g_0(x)$ of the NFSR is defined as

$$\begin{aligned} g_0(x) = {} & 1 + x^{17} + x^{20} + x^{28} + x^{35} + x^{43} + x^{47} + x^{52} + x^{59} + x^{65} + x^{71} + x^{80} + \\ & x^{43}x^{47} + x^{65}x^{71} + x^{20}x^{28}x^{35} + x^{47}x^{52}x^{59} + x^{17}x^{35}x^{52}x^{71} + \\ & x^{20}x^{28}x^{43}x^{47} + x^{17}x^{20}x^{59}x^{65} + x^{17}x^{20}x^{28}x^{35}x^{43} + x^{47}x^{52}x^{59}x^{65}x^{71} + \\ & x^{28}x^{35}x^{43}x^{47}x^{52}x^{59}. \end{aligned}$$

The NFSR feedback is disturbed by the output of the LFSR, so that the NFSR is governed by the recurrence:

$$\begin{aligned} b_{t+80} = s_t &+ b_{t+63} + b_{t+60} + b_{t+52} + b_{t+45} + b_{t+37} + b_{t+33} + b_{t+28} + b_{t+21} + b_{t+15} \\ &+ b_{t+9} + b_t + b_{t+63}b_{t+60} + b_{t+37}b_{t+33} + b_{t+15}b_{t+9} + b_{t+60}b_{t+52}b_{t+45} \\ &+ b_{t+33}b_{t+28}b_{t+21} + b_{t+63}b_{t+45}b_{t+28}b_{t+9} + b_{t+60}b_{t+52}b_{t+37}b_{t+33} \\ &+ b_{t+63}b_{t+60}b_{t+21}b_{t+15} + b_{t+63}b_{t+60}b_{t+52}b_{t+45}b_{t+37} \\ &+ b_{t+33}b_{t+28}b_{t+21}b_{t+15}b_{t+9} + b_{t+52}b_{t+45}b_{t+37}b_{t+33}b_{t+28}b_{t+21}. \end{aligned}$$

The contents of the two shift registers compose of the cipher state. The cipher output bit z_t^0 is derived from the current LFSR and NFSR states by a filter function $h_0(x_0, \cdots, x_4)$ as follows,

$$z_t^0 = b_t + h_0(s_{t+3}, s_{t+25}, s_{t+46}, s_{t+64}, b_{t+63}),$$

where the filter boolean function $h_0(x_0, \cdots, x_4)$ is defined as

$$\begin{aligned} h_0(x_0, \cdots, x_4) = x_1 &+ x_4 + x_0x_3 + x_2x_3 + x_3x_4 + x_0x_1x_2 + x_0x_2x_3 + x_0x_2x_4 \\ &+ x_1x_2x_4 + x_2x_3x_4. \end{aligned}$$

The Functions of Grain v1: For Grain v1, $k = 80$, $l = 64$. The LFSR feedback polynomial $f_1(x)$ is the same as $f_0(x)$. The NFSR feedback polynomial $g_1(x)$ is defined as

$$\begin{aligned} g_1(x) = 1 &+ x^{18} + x^{20} + x^{28} + x^{35} + x^{43} + x^{47} + x^{52} + x^{59} + x^{66} + x^{71} + x^{80} + \\ &+ x^{17}x^{20} + x^{43}x^{47} + x^{65}x^{71} + x^{20}x^{28}x^{35} + x^{47}x^{52}x^{59} + x^{17}x^{35}x^{52}x^{71} + \\ &+ x^{20}x^{28}x^{43}x^{47} + x^{17}x^{20}x^{59}x^{65} + x^{17}x^{20}x^{28}x^{35}x^{43} + x^{47}x^{52}x^{59}x^{65}x^{71} + \\ &+ x^{28}x^{35}x^{43}x^{47}x^{52}x^{59}. \end{aligned}$$

The filter function $h_1(x_0, \cdots, x_4)$ is also the same as $h_0(x_0, \cdots, x_4)$, but the cipher output bit z_t^1 is derived as

$$z_t^1 = \sum_{i \in \mathscr{A}_1} b_{t+i} + h_1(s_{t+3}, s_{t+25}, s_{t+46}, s_{t+64}, b_{t+63}),$$

where $\mathscr{A}_1 = \{1, 2, 4, 10, 31, 43, 56\}$.

The Functions of Grain-128: For Grain-128, $k = 128$, $l = 96$. The LFSR feedback polynomial $f_{128}(x)$ is a primitive polynomial of degree 128, defined as

$$f_{128}(x) = 1 + x^{32} + x^{47} + x^{58} + x^{90} + x^{121} + x^{128}.$$

The NFSR feedback polynomial $g_{128}(x)$ is as follows,

$$\begin{aligned} g_{128}(x) = 1 &+ x^{32} + x^{37} + x^{72} + x^{102} + x^{128} + x^{44}x^{60} + x^{61}x^{125} \\ &+ x^{63}x^{67} + x^{69}x^{101} + x^{80}x^{88} + x^{110}x^{111} + x^{115}x^{117}. \end{aligned}$$

The filter function $h_{128}(x_0, \cdots, x_8)$ is given as

$$h_{128}(x_0, \cdots, x_8) = x_0x_1 + x_2x_3 + x_4x_5 + x_6x_7 + x_0x_4x_8.$$

The cipher output bit z_t^{128} is derived as

$$z_t^{128} = \sum_{i \in \mathscr{A}_{128}} b_{t+i} + s_{t+93} \\ + h_{128}(b_{t+12}, s_{t+8}, s_{t+13}, s_{t+20}, b_{t+95}, s_{t+42}, s_{t+60}, s_{t+79}, s_{t+95}).$$

where $\mathscr{A}_{128} = \{2, 15, 36, 45, 64, 73, 89\}$.

2.2 Key Initialization

Before the generation of the cipher keystream, the cipher is initialized with the secret key and a selected IV. Let K_i be the i-th bit of the key K, $0 \leq i \leq k-1$. IV_i is the i-th bit of the IV, $0 \leq i \leq l-1$. The initialization of the key is done as follows. First load the NFSR with the key bits, $b_i = K_i$, $0 \leq i \leq k-1$, then load the first l bits of the LFSR with the IV, $s_i = IV_i$, $0 \leq i \leq l-1$. The remaining bits of the LFSR are filled with ones, i.e., $s_i = 1$, $l \leq i \leq k-1$. Then the cipher is clocked $2k$ times without producing any running key. It is noted that, during the key initialization, the output of the filter function is fed back both to the LFSR and the NFSR, see Fig.2. In the next section, we show that the feedback operation will produce some weak key and IV pairs which are called weak Key-IVs.

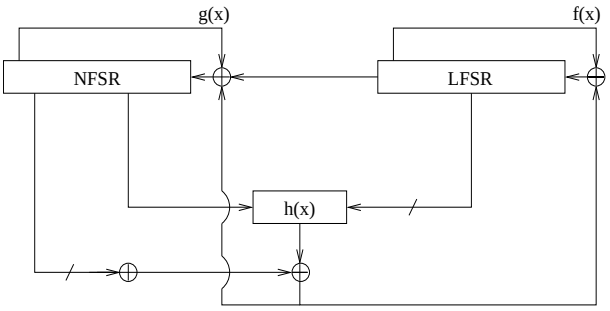

Fig. 2. The key initialization of Grain

3 The Existence of the Weak Key-IVs

Grain family uses a linear feedback shift register not only to ensure good statical properties, but also to guarantee a lower bound for the period of the keystream.

To introduce nonlinearity, a nonlinear feedback shift register is used together with a nonlinear filter, and the nonlinear filter takes inputs from both shift registers. However, if the initial states of the LFSR are all zeros after the initialization process, NFSR is the only active block of the cipher body. As well known, keystream sequences generated by a single NFSR are vulnerable to distinguishing attacks, such as short period, and linear approximation etc. Consequently, if the key and IV pair results in the all zero LFSR state, we define the key and IV pair as a **weak Key-IV**. In fact, in order to avoid weak Key-IVs, the designers set $k - l$ bits as ones which are loaded to the LFSR before key initialization. Unfortunately, our cryptanalysis reveals that, after $2k$ clocks, the state of the LFSR has the possibility to be zero state.

Now we take Grain v1 as an example to reveal the existence of the weak Key-IVs. Denote the internal state of the NFSR as $B_t = (b_t, b_{t+1}, \cdots, b_{t+79})$, and the internal state of the LFSR as $S_t = (s_t, s_{t+1}, \cdots, s_{t+79})$. Let $\mathscr{G}$ and $\mathscr{F}$ be the state transform functions as $B_{t+1} = \mathscr{G}(B_t)$ and $S_{t+1} = \mathscr{F}(S_t)$. Then the key initialization can be regarded as the process that B_0 and S_0 transfer into B_{160} and S_{160}. For a weak Key-IV, the state S_{160} is the zero state $(0, 0, \cdots, 0)$. Then our following purpose is to obtain an available (B_0, S_0) from known B_{160} and S_{160} in reverse. According to the structure of Grain v1, we present a simple algorithm to compute the weak Key-IVs.

Algorithm 1.

1. Set S_{160} be the zero state $(0, 0, \cdots, 0)$, and select B_{160} randomly.
2. From $t = 159$ to $t = 0$ do
   (a) Compute $z_t^1 = \sum_{i \in \mathscr{A}_1} b_{t+i} + h_1(s_{t+3}, s_{t+25}, s_{t+46}, s_{t+64}, b_{t+63})$.
   (b) Compute $s_t = z_t^1 + s_{t+80} + s_{t+62} + s_{t+51} + s_{t+38} + s_{t+23} + s_{t+13}$.
   (c) Compute $b_t = z_t^1 + b_{80+t} + s_t + P(B_t)$, where $P(B_t)$ is a expression of 79 variables $b_{t+1}, b_{t+2}, \cdots, b_{t+79}$.
3. For $j = 64, \cdots, 79$, if $s_j = 1$ always holds, terminate; else, go to step 1.

After executing the above algorithm 2^{20} times, we obtain 16 weak Key-IVs. Similarly, we can find the weak Key-IVs of Grain v0 and Grain-128. Some examples are illustrated in Table 1. The most significant bits correspond to the first bits K_0 and IV_0 in the weak Key-IV respectively.

Because of the feedback operation of filter function to LFSR and NFSR, the internal state with $2k$ bits of the NFSR and LFSR is uniformly distributed after the key initialization. Thus there are 2^l weak Key-IVs among total 2^{k+l} Key-IVs for any Grain version.

4 Distinguishing the Weak Key-IVs

In order to detect a weak Key-IV, we apply the second Walsh spectra to the NFSR, and obtain its best linear approximation.

Table 1. Weak Key-IVs of Grain Family.

Version	Grain v0	Grain v1
Key	0x6f22a2a70e1c363b62af	0xf57e358ecae6b3dc683d
IV	0x44b604a4d4479eb4	0x97652a7f1a112415
B_{160}	0xc2ced7db3189a9ad94b8	0xd99ea5abb8d0129212c7
S_{160}	0x00000000000000000000	0x00000000000000000000
Version	Grain-128	
Key	0xfd6af0ff0ad9bdad7037b91ef1b9cc13	
IV	0x014d3e274f8d3528ddad4310	
B_{160}	0xc1bc1c087a79b533f9018d230df2e744	
S_{160}	0x00000000000000000000000000000000	

Suppose that $x = (x_0, \cdots, x_{n-1})$, $\omega = (\omega_0, \cdots, \omega_{n-1}) \in GF(2)^n$. The dot production of x and ω is defined as

$$x \circ \omega = x_0\omega_0 + \cdots + x_{n-1}\omega_{n-1} \in \mathrm{G}F(2).$$

Given any value w, it is easy to compute the second Walsh spectra of boolean function $f(x)$ as follows:

$$S_{(f)}(\omega) = \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{x \circ \omega}.$$

It is well known that the linear approximation of function $f(x)$ can be characterized by Lemma 1.

Lemma 1. Given $x = (x_0, \cdots, x_{n-1})$, $\omega = (\omega_0, \cdots, \omega_{n-1}) \in \mathrm{GF}(2)^n$, and $f(x)$ is a boolean function, there are

$$\mathrm{Prob}(f(x) = \omega \circ x) = \frac{1 + S_{(f)}(\omega)}{2},$$

$$\mathrm{Prob}(f(x) \neq \omega \circ x) = \frac{1 - S_{(f)}(\omega)}{2}.$$

By searching all the values w, we can get the best linear approximation for the boolean function $f(x)$.

4.1 Distinguishing the Weak Key-IVs of Grain v0

It is obvious that for the Grain v0 with a weak Key-IV, the NFSR runs only by itself. So we can utilize the second Walsh spectra to obtain the best linear approximation of function $g_0(x)$. Firstly, we transform $g_0(x)$ into multi-variable form as $G_0(y_0, \cdots, y_{10})$, where $(y_0, \cdots, y_{10}) = (x^{80}, x^{71}, x^{65}, x^{59}, x^{52}, x^{47}, x^{43}, x^{35}, x^{28},$

x^{20}, x^{17}). Then we can obtain the maximum second Walsh spectra value of G_0 is

$$S_{(G_0)}(u) = \max_{\omega \in \mathrm{GF}(2)^{11}} S_{(G_0)}(\omega) = \frac{328}{2^{11}},$$

here $u = (1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1)$. Consequently, the NFSR can be approximated by the following linear recursion:

$$b_{t+80} = b_{t+63} + b_{t+52} + b_{t+45} + b_{t+37} + b_{t+28} + b_{t+21} + b_{t+15} + b_{t+9} + b_t, \quad (1)$$

and by Lemma 1, the probability that the recursion holds is

$$p_0 = \frac{1 + S_{(G_0)}(u)}{2} = \frac{1}{2} + \frac{164}{2^{11}}. \quad (2)$$

For the Grain v0 with a weak Key-IV, the internal states of LFSR are all zeroes. Consequently, the equations s_{t+3}=s_{t+25}=s_{t+46} =$s_{t+64} = 0$ always hold. Then the filter function h_0 is simplified as

$$h_0(s_{t+3}, s_{t+25}, s_{t+46}, s_{t+64}, b_{t+63}) = b_{t+63}.$$

Therefore, the keystream sequence satisfies the following equation:

$$z_t^0 = b_t + b_{t+63}. \quad (3)$$

Provided that the equation (1) holds at both the clock t and clock $t + 63$, then,

$$\sum_{i \in \mathscr{B}_0} z_{t+i}^0 = \sum_{i \in \mathscr{B}_0} b_{t+i} + \sum_{i \in \mathscr{B}_0} b_{t+63+i}, \quad (4)$$

where $\mathscr{B}_0 = \{0, 9, 15, 21, 28, 37, 45, 52, 63, 80\}$.

Because the equation (1) holds with a probability bias $\frac{41}{2^9}$, the following equation also holds with a probability bias.

$$\sum_{i \in \mathscr{B}_0} b_{t+i} + \sum_{i \in \mathscr{B}_0} b_{t+63+i} = 0, \quad (5)$$

where $\mathscr{B}_0 = \{0, 9, 15, 21, 28, 37, 45, 52, 63, 80\}$.

By Piling-up Lemma and the probability (2), the equation (5) holds with probability

$$\frac{1}{2} + 2 \times \frac{164}{2^{11}} \times \frac{164}{2^{11}} = \frac{1}{2} + 2^{-6.3}.$$

So, if the keystream sequence is generated by Grain v0 with a weak Key-IV, the equation (5) with it's probability bias $2^{-6.3}$ consists of a distinguisher. Thus, to distinguish the Grian v0 with a weak Key-IV only need $2^{6.3 \times 2} = 2^{12.6}$ keystream bits, and the time complexity is about $2^{12.6} \times 10 \approx 2^{15.8}$ XOR operations. For Grain v0, $k = 80$ and $l = 64$, so there are 2^{64} weak Key-IVs among 2^{144} Key-IVs.

4.2 Distinguishing the Weak Key-IVs of Grain v1

Similarly, we can distinguish a weak Key-IV of the Grain v1 as above. Firstly, transform $g_1(x)$ into multivarible form as $G_1(y_0, \cdots, y_{12})$, where $(y_0, \cdots, y_{12}) = (x^{80}, x^{71}, x^{66}, x^{59}, x^{52}, x^{47}, x^{43}, x^{35}, x^{28}, x^{20}, x^{18}, x^{17}, x^{65})$, the maximum second Walsh spectra value of G_1 is

$$S_{(G_1)}(u) = \max_{\omega \in \mathrm{GF}(2)^{13}} S_{(G_1)}(\omega) = \frac{1312}{2^{13}},$$

here $u = (1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0)$. The NFSR can be approximated by the following linear recursion:

$$b_{t+80} = b_{t+62} + b_{t+60} + b_{t+52} + b_{t+45} + b_{t+37} + b_{t+28} + b_{t+21} + b_{t+14} + b_t, \quad (6)$$

and from Lemma 1, the probability that the recursion holds is

$$p_1 = \frac{1 + S_{(G_1)}(u)}{2} = \frac{1}{2} + \frac{656}{2^{13}}. \quad (7)$$

For a weak Key-IV of Grain v1, the keystream sequence is computed as

$$z_t^1 = \sum_{i \in \mathscr{B}_1} b_{t+i}, \quad (8)$$

where $\mathscr{B}_1 = \{1, 2, 4, 10, 31, 43, 56, 63\}$. From (6) and (8), a distinguisher can be established as follows:

$$\sum_{i \in \mathscr{B}_2} z_{t+i}^1 = \sum_{i \in \mathscr{B}_2} \sum_{j \in \mathscr{B}_1} b_{t+i+j} = 0, \quad (9)$$

where $\mathscr{B}_2 = \{0, 14, 21, 28, 37, 45, 52, 60, 62, 80\}$.

According to Piling-up Lemma and the probability (7), the equation (9) will holds with probability

$$\frac{1}{2} + 2^7 \times (\frac{656}{2^{13}})^8 = \frac{1}{2} + 2^{-22.1}.$$

Thus, to detect a weak Key-IV of Grain v1 needs about $2^{44.2}$ keystream bits, and the time complexity is about $2^{47.5}$ XOR operations. For Grain v1, $k = 80$ and $l = 64$, the number of weak Key-IVs is about 2^{64} among 2^{144} Key-IVs.

4.3 Distinguishing the Weak Key-IVs of Grain-128

For Grain-128, the nonlinear terms of $g_{128}(x)$ implies a Bent Boolean function $g_{128}^N(y_0, \cdots, y_{13})$. Therefore, given any $\omega \in \mathrm{GF}(2)^{14}$, the second Walsh spectra value $S_{(g_{128}^N)}(\omega)$ will always be $\pm\frac{128}{2^{14}}$. i.e., the Hamming distance between g_{128}^N and its any linear approximation function Lg_{128}^N is all the same. The NFSR can be approximated by the following linear recursion:

$$\begin{aligned} b_{t+128} = b_{t+96} + b_{t+91} + b_{t+56} + b_{t+26} + b_t + Lg_{128}^N(&b_{t+84}, b_{t+68}, b_{t+67}, b_{t+3}, \\ &b_{t+65}, b_{t+61}, b_{t+59}, b_{t+27}, b_{t+28}, b_{t+20}, b_{t+18}, b_{t+17}, b_{t+13}, b_{t+11}), \end{aligned}$$

here $Lg_{128}^{N}(b_{t+84}, b_{t+68}, ..., b_{t+11})$ is any linear expression of 14 variables (b_{t+84}, b_{t+68}, b_{t+67}, b_{t+3}, b_{t+65}, b_{t+61}, b_{t+59}, b_{t+27}, b_{t+28}, b_{t+20}, b_{t+18}, b_{t+17}, b_{t+13}, b_{t+11}). If a keystream sequence is generated by Grain-128 with a weak Key-IV, the output keystream z_t^{128} can be represented as

$$z_t^{128} = \sum_{i \in \mathscr{A}_{128}} b_{t+i}. \tag{10}$$

Furthermore, a distinguisher can be constructed as follows:

$$\sum_{i \in \mathscr{B}_{128}} z_{t+i}^{128} = \sum_{i \in \mathscr{B}_{128}} \sum_{j \in \mathscr{A}_{128}} b_{t+i+j} = 0, \tag{11}$$

where $\mathscr{B}_{128} = \{0, 26, 56, 91, 96, 128\} \cup \mathscr{C}_{128}$, $\mathscr{C}_{128}$ is an any subset of $\{3, 11, 13, 17, 18, 20, 27, 28, 59, 61, 65, 67, 68, 84\}$.

It is easy to know that the equation (11) will holds with probability

$$\frac{1}{2} \pm 2^6 \times (\frac{64}{2^{14}})^7 = \frac{1}{2} \pm 2^{-50}.$$

Because the number of the subset $\mathscr{C}_{128}$ is 2^{14}, to distinguish a weak Key-IV of Grain-128, $2^{50 \times 2 - 14} = 2^{86}$ keystream bits are required, and the time complexity is about $2^{104.2}$ XOR operations. For Grain-128 with $k = 128$ and $l = 96$, the number of weak key-IVs is about 2^{96} among total 2^{224} Key-IVs.

5 Recovering the NFSR Initial State of the Weak Key-IV

A weak Key-IV can be distinguished means that the LFSR initial state of the weak Key-IV is the zero state. Consequently, the next task is to recover the NFSR initial state.

For Grain v0, we firstly recall the recovering technique in [3]. To recover all the bits of the NFSR initial state after the LFSR initial state is recovered, they introduced a technique which consists of building chains of keystream bits. The knowledge of the LFSR removes the nonlinearity of the output, and each keystream bit z_t^0 can be expressed by one of the following four equations depending on the initial state of the LFSR:

$$\begin{aligned} z_t^0 &= b_t, \\ z_t^0 &= b_t + 1, \\ z_t^0 &= b_t + b_{t+63}, \\ z_t^0 &= b_t + b_{t+63} + 1. \end{aligned}$$

The equation involving only one bit allows them to instantly recover the value of the corresponding bit of the initial state. This is the core the technique works efficiently. However, if the LFSR initial state is all zeroes, only the equation $z_t^0 = b_t + b_{t+63}$ holds. There are two bits involved in the equation, then the

technique can not be applied to the Key-IVs. This implies that there are 2^{64} Key-IVs can not be recovered successfully by the technique in [3]. The resolvent is to apply the algebraic attack as follows.

Solving system of multivariate polynomial equations is NP-complete even if all the equations are quadratic. Only a limited number of distinct methods exist for solving system of polynomial equations over a finite field and they can be grouped as follows: Gröbner bases method, XL method, Zhuang-Zi method. The first one is the most important method of the three. It was introduced by Buchberger in the 1970s and it has been refined since then [2]. F4 and F5 algorithm [6, 7], proposed by Faugère, are the fastest implementations of algorithm for finding Gröbner bases so far.

The internal state of Grain v1 is of 160 bits, with 80 bits each of LFSR and NFSR. Some internal states, at each clock, are updated with nonlinear expressions. And succeeding equations which involve these state bits have even more higher degree. We can know that after almost 80 equations, the degree of the equations become as high as 160, which is the maximum possible degree here. Solving such high degree equations for all 160 variables is beyond the available resources. In [1], a number of experiments are performed to find out the maximum number of bits that can be recovered while other are guessed. Experiments show that out of all 160 bits no more than 77 bits can be recovered, while remaining 83 are guessed, and only the situation when all bits of LFSR are guessed and 3 last bits of NFSR are guessed, results in solution of the equations to give 77 unknown bits. For Grain-128, after a number of experiments with different number of guessed bits, it is construed that not more than 64 bits can be recovered from the algebraic equations within the available resources, and 128 LFSR initial state bits must be guessed. Furthermore, M. Afzal and A. Masood generated algebraic equations of Grain v1 and Grain-128 in Maple 10, and solved the nonlinear equations with Magma V 2.13-5 [13] on a PC with CPU at 1.73 GHz and 1 GB RAM. The summary of simulation results of algebraic analysis of Grain v1 and Grain-128 is illustrated in Table 2.

Table 2. Simulation results of algebraic analysis of two versions of Grain

Version	Unknown bits	Degree of $g(x)$	Degree of $h(x)$	No. of bits guessed	No. of bits recovered	Time to find solution	Keystream bits used
Grain v1	160	6	3	83	77	0.204 sec	150
Grain-128	256	2	3	192	64	0.906 sec	100

For the weak Key-IV of Grain v1, the LFSR initial state is the zero state, and the degree of $h(x)$ degenerates into 1, so only a total of 80 internal state bits are unknown. Similarly, for the weak Key-IV of Grain-128, the degree of $h(x)$ becomes 0, and there are 128 unknown NFSR initial state bits. So, we can directly utilize the result in Table 2, and the attack result of the weak Key-IVs can be illustrated in Table 3.

Table 3. Simulation results of algebraic analysis of the weak Key-IVs of Grain

Version	Unknown bits	Degree of $g(x)$	Degree of $h(x)$	No. of bits guessed	No. of bits recovered	Time to find solution	Keystream bits used
Grain v1	80	6	1	3	77	0.204 sec	150
Grain-128	128	2	0	64	64	0.906 sec	100

Suppose that one operation acts per frequency of the CPU, $1.73 \times 10^9 = 2^{29.9}$ operations execute by one second for 1.73 GHz PC. From Table 3, we can conclude that the weak Key-IVs of Grain v1 can be broken with $2^{29.9} \times 0.204 \times 2^3 = 2^{30.7}$ operations and 150 keystream bits, and the weak Key-IVs of Grain-128 can be broken with $2^{29.9} \times 0.906 \times 2^{64} = 2^{93.8}$ operations and 100 keystream bits. The result of Grain v0 is similar to that of Grain v1.

6 Self-sliding Attack on the Weak Key-IVs of Grain-like Structure

In this section, we try to generalize the attack against the weak Key-IVs of Grain-like structure which consists of a LFSR, a NFSR and a filter like Grain family. The attack can be divided into two steps:

- Utilizing the best linear approximation of $g(x)$ and $h(x)$, distinguish the weak Key-IVs.
- Recover the internal state of the NFSR by the algebraic attack.

For Grain-like structure, the keystream bit z_t can be represented as the output of the boolean function $h'(x)$ which input variables taken from both the LFSR and NFSR, i.e., $z_t = h'(x)$. Let the function $A_g(x)$ be a linear approximation of the function $g(x)$, and $A_{h'}(x)$ be a linear approximation of the the function $h'(x)$. Suppose that $w_N(A_g)$ is the the number of variables taken from the NFSR, and $w_N(A_{h'})$ also is the number of variables from the NFSR. Then Lemma 2 is given as follows.

Lemma 2. There always exists a linear relation in terms of bits from the state of the LFSR and the keystream, which have the bias:

$$\epsilon = 2^{(w_N(A_{h'})+w_N(A_g)-1)} \cdot \epsilon_g^{w_N(A_{h'})} \cdot \epsilon_{h'}^{w_N(A_g))},$$

where $\Pr\{A_g(\cdot) = g(\cdot)\}=\frac{1}{2} + \epsilon_g$ and $\Pr\{A_{h'}(\cdot) = h'(\cdot)\}=\frac{1}{2} + \epsilon_{h'}$.

See Theorem 1 in Section 3.2 in [3].

Because in the case of Grain v0 the functions $g(x)$ and $h'(x)$ are improperly chosen, Grain v0 is broken. There are two ways to avoid the attack on Grain v0, one is to increase the number $w_N(A_g)$ and $w_N(A_{h'})$, the other is to decrease the

bias ϵ_g and $\epsilon_{h'}$. The bias ϵ in Grain v1 and Grain-128 is low enough to the strong Key-IVs, however it is still vulnerable to the weak Key-IVs discussed above.

In a general way, let $g(x)$ and $h'(x)$ be random functions such that ϵ is extremely close to 0. In this case, any linear approximation distinguishing attack is disabled. Considering the LFSR initial state of the weak Key-IVs is the zero state, the keystream output bits only depend on the initial state of the NFSR.

The cycle structure of random functions was studied by Knuth in connection with random number generators and by Brent and Pollard in connection with factorization. Knuth obtains an average cycle length, for a random function over L values, of $(\frac{\pi L}{8})^{\frac{1}{2}} + \frac{1}{3}$ and an average tail length of $(\frac{\pi L}{8})^{\frac{1}{2}} - \frac{2}{3}$. The $L^{\frac{1}{2}}$ relationship is related to the 'birthday problem'. Consequently, we can utilize the self-sliding attack to distinguish the weak Key-IVs. The average data complexity is about

$$(\frac{\pi L}{8})^{\frac{1}{2}} + \frac{1}{3} + (\frac{\pi L}{8})^{\frac{1}{2}} - \frac{2}{3} = 2(\frac{\pi L}{8})^{\frac{1}{2}} - \frac{1}{3}.$$

Concretely, the average data complexities are $2^{39.7}$ and $2^{63.7}$ with corresponding to Grain-like structure with k being 80 and 128.

To avoid the self-sliding attack, the cycle of $g(x)$ should be 2^k, i.e., the sequence generated by $g(x)$ is a M-sequence. N. G. de Bruijn proved that there are $2^{2^{k-1}-k}$ different M-sequences among total 2^{2^k} boolean functions. Thus, randomly choose a k variables function, the probability that its cycle will be 2^k is

$$\frac{2^{2^{k-1}-k}}{2^{2^k}} = 2^{-2^{k-1}-k}.$$

This implied that the self-sliding attack may be applied with extremely high probability in the case of $g(x)$ being a random function. Thus, the simple resolvent is to modify the key initialization process which should guarantee the initial LFSR state always being no-zero.

7 Conclusion

In this paper, we described a distinguishing attack against the weak Key-IVs of Grain family. Utilizing the second Walsh spectra of the nonlinear functions, we present that there are $2^{64}/2^{64}/2^{96}$ weak Key-IVs, and to distinguish a weak Key-IV needs $2^{12.6}/2^{44.2}/2^{86}$ keystream bits and about $2^{15.8}/2^{47.5}/\ 2^{104.2}$ operations for Grain v0, Grain v1 and Grain-128 respectively.

Secondly, we apply algebraic attacks against the Key-IVs of the Grain family, and show that the weak Key-IVs can be broken in about 2 seconds utilizing about 150 keystream bits for Grain v0 and Grain v1. To break the weak Key-IVs of Grain-128, 100 keystream bits and $2^{93.8}$ operations are required. Furthermore, we discuss the periods of the weak Key-IVs which lead in self-sliding attacks against the Grain-like structure. Our results show that the key initialization process of Grain family should be modified.

References

1. M. Afzal, and A. Masood. Algebraic Cryptanalysis of A NLFSR Based Stream Cipher. Information and Communication Technologies: From Theory to Applications, ICTTA 2008, pp. 1-6, 2008.
2. B. buchberger, Gröbner Base: An Algorithm Method in Polynomial Ideal Theory, Multideimension System Theory. Dordrecht, pp. 184-232, 1985.
3. C. Berbain, H. Gilbert, and A. Maximov. Cryptanalysis of Grain. In M. J. B. Robshaw Editor, FSE 2006, LNCS 4047, pp. 15-29, 2006.
4. C. D. Cannière, ö. Küçük, and B. Preneel. Analysis of Grains Initialization Algorithm. In S. Vaudenay Editor, AFRICACRYPT 2008, LNCS 5023, pp. 276-289, 2008.
5. ECRYPT. eSTREAM: ECRYPT Stream Cipher Project, IST-2002-507932. Available at http://www.ecrypt.eu.org/stream/.
6. J. C. Faugère, A New Efficient Algorithm for Computing Gröbner Bases (F4), Journal of Pure and Applied Algebra. Vol.139, N0.1-3, pp. 61-88, June,1999.
7. J. C. Faugère, A New Efficient Algorithm for Computing Gröbner Bases without Reduction to Zero (F5), International Sysmposium on Symbolic and Algebraic Computation-ISSAC'02, pp. 75-83, ACM Press, 2002.
8. M. Hell, T. Johansson, A. Maximov, and W. Meier. The Grain Family of Stream Ciphers. In M. Robshaw and O. Billet Editors, New Stream Cipher Designs, LNCS 4986, pp. 179-190, 2008.
9. M. Hell, T. Jonasson, and W. Meier. Grain- A Stream Cipher for Constrained Enviroments. ECRYPT Stream Cipher Project Report 2005/001, 2005. Available at http://www.ecrypt.eu.org/stream.
10. O. ö. Küçük, Slide Resynchronization Attack on the Initialization of Grain 1.0. ECRYPT Stream Cipher Project Report 2006/044, 2006. Available at http://www.ecrypt.eu.org/stream.
11. S. Khazaei, M. Hassanzadeh and M.Kiaei. Distinguishing Attack on Grain. ECRYPT Stream Cipher Project Report 2005/071, 2005. Available at http://www.ecrypt.eu.org/stream.
12. Y. Lee, K. Jeong, J. Sung, and S. Hong. Related-Key Chosen IV Attacks on Grain-v1 and Grain-128. In Y. Mu, W. Susilo, and J. Seberry (Eds.), ACISP 2008, LNCS 5107, pp. 321-335, 2008.
13. Magma Computational Algebra System available at http://magma.maths.usyd.edu.au/.
14. NESSIE. New European Schemes for Signatures, Integrity, and Encryption. Available at http://www.cryptonessie.org.