I.J. Computer Network and Information Security, **2014, 6, 34-40**
Published Online May 2014 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2014.06.05

A Detailed Analysis of Grain family of Stream Ciphers

Mohammad Ubaidullah Bokhari
Aligarh Muslim University, Aligarh
Email: mubokhari@gmail.com

Shadab Alam
Aligarh Muslim University, Aligarh
Email: s4shadab@gmail.com

Syed Hamid Hasan
King Abdulaziz University, Kingdom of Saudi Arabia
Email: shh786@hotmail.com

Abstract—Hardware based ciphers are most suitable for resource constrained environments to provide information security and confidentiality. Grain is one such hardware based synchronous stream cipher. The motive of this study is to present a comprehensive survey and review of Grain family of stream ciphers that is one of the portfolio candidates in the hardware based category of eSTREAM. Security features and different attacks on these ciphers have been studied in this paper to analyze the strengths and weaknesses of these designs.

Index Terms—Information Security, Cryptography, eSTREAM, Stream Cipher, Grain.

I. Introduction

In the last decade, we have witnessed an explosive growth of the digital data. On every walk of our like is becoming increasingly dependent on digital data and communication. The life is becoming so fast that there is no place for the manual or the hard bind data transfer. Internet and data communication technologies have become an integral part of our life. Without these technologies we cannot assume the life to go on, but these public networks and wireless medium of data communication are very much susceptible to be hacked or compromised by unauthorized users. What will be the cost of such leakage of the data; we cannot think when it is concerned with financial institutions or defense services. Therefore, these information sharing or data communication technologies should be adequately secure and confidential. Confidentiality means that the information should be out of reach to others except who are authorized to know it.

Cryptography is the one of the oldest and major techniques involved with security and confidentiality of the data. Cryptographic algorithms are classified into two categories, Symmetric key and Asymmetric key based on keys used for encryption and decryption. Symmetric key algorithms use the same key for encryption and decryption, but asymmetric key algorithms use different key for encryption and decryption. Stream ciphers are the part of symmetric key cryptography, which has recently attracted the attention of the cryptographers and researchers. Stream ciphers operate on bit by bit level, but block ciphers operate on a fixed size of blocks of data. The other class of symmetric primitive is Block cipher which has been thoroughly studied and standardized.

AES is the standard block cipher which is widely used, but there remain many applications were stream ciphers are preferred choice and cannot be ignored. In the applications where a high rate of throughput and low hardware and memory complexity is required, stream ciphers are the natural choice due to its low complexity and high efficiency. Stream ciphers operate on individual symbols with time varying transformations against the design of block ciphers which operate on blocks of symbols of fixed size with fixed transformations [1].

Stream ciphers try to work like one time pad (OTP) that is the only theoretically unbreakable cipher. Even with these advantages, the stream cipher designs have not been fully evolved and no standard design exists for stream ciphers. The eSTREAM project has tried to standardize the stream ciphers to a great extent and generated an interest in this field of cryptography. Grain is one of the submitted designs for eSTREAM. In this paper, we have tried to study the detailed design of Grain stream cipher and its subsequent versions and different cryptanalytic attacks on these stream cipher designs. The Grain V1 is a profile 2 stream cipher in the recently published eSTREAM portfolio by Ecrypt.

Section II defines the stream ciphers and its advantage and when and where they are suitable for applications. Section III and IV define the Grain family of stream ciphers and the general structure of the cipher design. Section V defines the key initialization process that takes place before actual keystream is generated for encryption. Section VI defines the different members of the Grain family of ciphers, feedback and update functions used in

these ciphers and attacks mounted on these ciphers. In section VII the various members of the Grain stream cipher have been compared on the basis of their software as well as hardware performance and other functions used in the design of these ciphers and in last the conclusion of this study has been presented.

II. Stream cipher, its properties and advantages

Symmetric key ciphers are classified into two categories; Block Cipher and Stream Cipher. A stream cipher is an important class of symmetric key cipher. Unlike Block cipher, which use fixed cryptographic transformations on block of characters, Stream cipher to encrypt single characters of plaintext one by one with time varying transformations. As the stream ciphers encrypt individual digits, it takes less buffer memory, less complex hardware circuitry and is comparatively faster than block ciphers.

Block cipher requires no memory, but stream cipher requires memory for the storage of the current state of function, which is being used for further encryption. This is the reason why the same bit is encrypted differently in case of stream ciphers when enciphered again and again, but that is not the case in block ciphers. AES in Counter Mode or Output Feedback Mode can also be used as stream cipher and any stream ciphers must be able to be more efficient than these block cipher modes of operation to be used in any practical application.

Shamir in his popular invited talk [2] “Stream Ciphers: Dead or Alive” and Babbage in his invited talk [3] “Stream Ciphers - What does industry want?” at state of the art of stream ciphers workshop in 2004 clearly identified some areas where stream ciphers have an edge over block ciphers.

These are the some areas where stream ciphers can be useful:

1. Stream ciphers have an edge over block ciphers where hardware resources are limited and less complex circuits are required like RFID tags and smart cards.
2. Stream ciphers can be useful in cases where very high speed throughput is required like multi gigabit communication channels.
3. Stream ciphers are also desirable where zero error propagation is required like radio communication, due to no error propagation in case of synchronous stream ciphers or limited error propagation in case of an asynchronous stream cipher.
4. Stream ciphers are also desirable where the length of the message cannot be predetermined and smaller input/output delay is required as in the case of GSM communication.

These are the few areas where stream ciphers have a clear edge over block ciphers due to its efficiency and speed.

III. Grain family of stream ciphers

Specific cryptographic primitives are required for resource constrained environments for information security and hardware based stream ciphers are most suitable for this purpose. Grain family of stream ciphers that is one of the portfolio ciphers in the hardware based category of eSTREAM is one of the cipher designs for such applications.

The original version of Grain referred as Grain V0 [4] was submitted to eSTREAM project [5] in the hardware category of stream ciphers. The grain V0 design was weak and it was susceptible to serious attacks. This design was tweaked and a new version of Grain called Grain V1 [6] was presented. Both of these versions of Grain used 80 bit key and 64 bit IV with an internal state of 160 bits. Grain was designed initially for security level of 280.

But due to rapid technical advancement in the field of hardware technology and speed of hardware, 80 bit ciphers are not found to be secure enough and susceptible to exhaustive key search attack. Therefore, it was desirable to have at least 128 bit security and to meet this requirement Grain 128 [7] was proposed by the designers of the Grain. Grain 128 uses 128 bit keys and 96 bit IV. In view of some cryptanalytic attacks on Grain 128, a new version of Grain 128 was introduced that also incorporate authentication named as Grain 128a. The new cipher was designed to overcome the existing weaknesses of Grain 128 and provide authentication when needed, otherwise behave similar to Grain 128 cipher. The new design was named Grain 128a [8] where "a" represent authentication. In this way there are four members in Grain family of ciphers, namely Grain V0, Grain V1, Grain 128, and Grain 128a.

IV. Design specifications of grain stream ciphers

The basic building blocks of all four variants are same and these use one Non Linear Feedback Shift Register (NFSR) and one Linear Feedback Shift Register (LFSR) with modifications in their feedback functions for different variants of this family. Grain family of ciphers is a bit oriented synchronous stream cipher.

The general structure of Grain family of stream cipher is given in Fig 1.

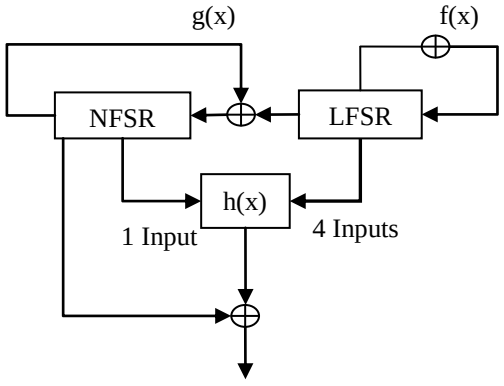

Fig 1: Overview of design blocks in Grain

The NFSR is updated with function g (x) and LFSR is updated with a function f(x). For keystream generation, 1 input is taken from NFSR and 4 inputs from LFSR and passed to the boolean function h(x) that gives a one bit output. That one bit output is again masked with the first bit of the NFSR to generate a keystream that will xored with the plaintext to generate the ciphertext.

V. Key Initialization of Grain

The cipher has to be initiated before it actually generates key streams. The secret key is loaded in the NFSR and first 64 in case of 80 bit ciphers and first 96 bits in case of 128 bit ciphers are loaded with the IV's [9]. The remaining vacant bit positions are filled with all ones. If the key size is K then the cipher is clocked 2K times without producing any keystream. The output of the filter function is fed back into both the shift registers. The logic behind clocking the cipher 2K times is that all the previously stored values before initialization phase from shift registers will be flushed out and only random values will be in the both shift registers.

Later on, after the observation by Kucuk [10] the designers chose to fill the last 31 bits of LFSR by ones and rightmost bit with zero to counter this attack in Grain 128a. The key initialization process has been shown in Fig 2.

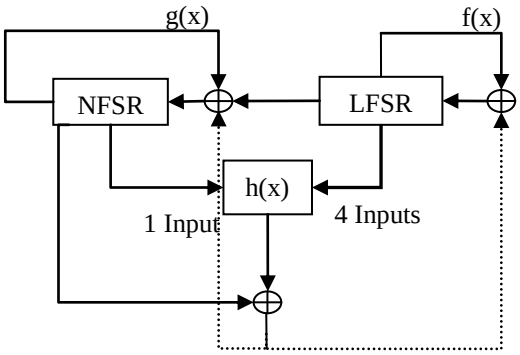

Fig 2: Key Initialization of Grain

VI. Members of Grain Family of Stream Ciphers

There are four members of Grain family of stream ciphers. In this section we have discussed the design specifications, feedback polynomials and different attacks against these ciphers.

A. Grain V_0:

Grain V0 was the first design that was submitted to eSTREAM in the hardware profile of stream ciphers. Grain V0 is a 80 bit stream cipher that uses two feedback shift registers; one LFSR and one NFSR of 80 bits each and with internal state of 160 bits that has been assumed to be secure against all the attacks with complexities less than O (280).

The feedback polynomial of LFSR used to update the register is defined as:

$$\boldsymbol{f(x)} = 1 + x^{18} + x^{29} + x^{42} + x^{57} + x^{67} + x^{80}$$

It is a irreducible primitive polynomial of degree 80.
The update function of LFSR is defined as:

$$\boldsymbol{s_{i+80}} = s_i + s_{i+13} + s_{i+23} + s_{i+38} + s_{i+51} + s_{i+62}$$

Feedback polynomial of NFSR

$$\boldsymbol{g(x)} = 1 + x^{17} + x^{20} + x^{28} + x^{35} + x^{43} + x^{47} + x^{52} + x^{59} + x^{65} + x^{71} + x^{80} + x^{17}x^{20} + x^{43}x^{47} + x^{65}x^{71} + x^{20}x^{28}x^{35} + x^{47}x^{52}x^{59} + x^{17}x^{35}x^{52}x^{71} + x^{20}x^{28}x^{43}x^{47} + x^{17}x^{20}x^{59}x^{65} + x^{17}x^{20}x^{28}x^{35}x^{43} + x^{47}x^{52}x^{59}x^{65}x^{71} + x^{28}x^{35}x^{43}x^{47}x^{52}x^{59}$$

And hence the update function of NFSR is defined as:

$$\boldsymbol{b_{i+80}} = s_i + b_{i+63} + b_{i+60} + b_{i+52} + b_{i+45} + b_{i+37} + b_{i+33} + b_{i+28} + b_{i+21} + b_{i+15} + b_{i+9} + b_i + b_{i+63}b_{i+60} + b_{i+37}b_{i+33} + b_{i+15}b_{i+9} + b_{i+60}b_{i+52}b_{i+45} + b_{i+33}b_{i+28}b_{i+21} + b_{i+63}b_{i+45}b_{i+28}b_{i+9} + b_{i+60}b_{i+52}b_{i+37}b_{i+33} + b_{i+63}b_{i+60}b_{i+21}b_{i+15} + b_{i+63}b_{i+60}b_{i+52}b_{i+45}b_{i+37} + b_{i+33}b_{i+28}b_{i+21}b_{i+15}b_{i+9} + b_{i+52}b_{i+45}b_{i+37}b_{i+33}b_{i+28}b_{i+21}$$

The filter function *h(x)* is a Boolean function that takes five inputs and gives a single output, has been given as:

$$\boldsymbol{h(x)} = x_1 + x_4 + x_0x_3 + x_2x_3 + x_3x_4 + x_0x_1x_2 + x_0x_2x_3 + x_0x_2x_4 + x_1x_2x_4 + x_2x_3x_4$$

Where the variables x_0, x_1, x_2, x_3 and x_4 correspond to the tap positions s_{i+3}, s_{i+25}, s_{i+46}, s_{i+64} and b_{i+63} respectively
Keystream function is defined as:

$$z_t = x_t \oplus h(y_{t+3}, y_{t+25}, y_{t+46}, y_{t+64}, x_{t+63})$$

Attacks on Grain V_0:

A distinguishing attack against Grain V0 was mounted by Khazaei, Hassanzadeh and Kiaei [11] that uses the concepts of linear sequential circuit approximation method given by Golic. This attack also requires a preprocessing phase to compute the trinomial multiples of some primitive polynomials of degree 80 and requires time and memory complexity of O (2^{40}). This distinguishing attack can distinguish a Grain output sequence from a purely random one with a complexity of O ($2^{61.4}$).

The second attack was presented by Barbein, Gilbert and Maximov [12] that is a key recovery attack against Grain V0. In this attack first of all, the linear approximation method is used to derive the LFSR bits and these LFSR bits are further utilized to recover the initial state of NFSR and knowledge of key. This attack requires 238 keystream bits and computational complexity of O (243) to recover the key.

In order to thwart these attacks and strengthen the designers of Grain have proposed a new design Grain V_1 and submitted it to eSTREAM.

B. Grain V_1:

The new version of Grain called as Grain V1 also has the similar design specifications as in Grain V0 and it is also a 80 bit stream cipher that uses two shift registers,

one NFSR and one LFSR of 80 bits each and give an internal state of 160 bits.

The feedback polynomial of LFSR was retained same as in Grain V0 but the feedback polynomial and update function of NFSR was slightly modified to overcome the weaknesses of Grain V0.

The new feedback polynomial *g1(x)* of NFSR is defined as:

$$g_1(x) = 1 + x^{18} + x^{20} + x^{28} + x^{35} + x^{43} + x^{47} + x^{52} + x^{59} + x^{65} + x^{71} + x^{80} + x^{17}x^{20} + x^{43}x^{47} + x^{65}x^{71} + x^{20}x^{28}x^{35} + x^{47}x^{52}x^{59} + x^{17}x^{35}x^{52}x^{71} + x^{20}x^{28}x^{43}x^{47} + x^{17}x^{20}x^{59}x^{65} + x^{17}x^{20}x^{28}x^{35}x^{43} + x^{47}x^{52}x^{59}x^{65}x^{71} + x^{28}x^{35}x^{43}x^{47}x^{52}x^{59}$$

And hence the new update function of NFSR as per the new feedback polynomial of NFSR is defined as:

$$b_{i+80} = s_i + b_i + b_{i+9} + b_{i+14} + b_{i+21} + b_{i+28} + b_{i+33} + b_{i+37} + b_{i+45} + b_{i+52} + b_{i+60} + b_{i+62} + b_{i+9}b_{i+15} + b_{i+33}b_{i+37} + b_{i+60}b_{i+63} + b_{i+21}b_{i+28}b_{i+33} + b_{i+45}b_{i+52}b_{i+60} + b_{i+15}b_{i+21}b_{i+60}b_{i+63} + b_{i+33}b_{i+37}b_{i+52}b_{i+60} + b_{i+9}b_{i+28}b_{i+45}b_{i+63} + b_{i+9}b_{i+15}b_{i+21}b_{i+28}b_{i+33} + b_{i+37}b_{i+45}b_{i+52}b_{i+60}b_{i+63} + b_{i+21}b_{i+28}b_{i+33}b_{i+37}b_{i+45}b_{i+52}$$

The filter function is same as Grain V_0 but the keystream function was slightly modified.

The new keystream function is defined as :

$$z_i = \sum_{k\in A}^{n} b_{i+k} + h(s_{i+3}, s_{i+25}, s_{i+46}, s_{i+64}, b_{i+63})$$

Where A = {1, 2,4,10, 31, 43, 56}

Attacks on Grain V_1:

Canniere, Kucuk and Preneel [13] mounted an attack on Grain V_1 by using a weakness in initialization algorithm. This attack was an extension of the work carried out by Kucuk in [10]. These two attacks have exploited the sliding property of the Grain V_1 that is due to similarity in key initialization and key generation processes. The attackers have claimed to reduce the attack complexity by half of the exhaustive key search attack.

Lee et al [14] have extended and proposed a sophisticated attack by exploiting the same weakness of related key in Grain V1. This attack is a key recovery attack that recovers the key with 222.59 chosen IVs, 226.29 keystream bits and 222.90 computations.

Bjorstad also proposed TMTO attack [15] using known keystream bits of O (253.5) and time and memory complexity of O (271) but this attack was of no practical significance except it shows some weakness in design.

Recently Dynamic Cube attack [16] was also proposed against the Grain V1 by Rahimi et al. This attack can fully recover the 80 bit key if initialization rounds are reduced to 100 with the computational complexity of 248.

C. Grain 128:

If the key size of a stream cipher is K then a Time Memory Tradeoff attack can be mounted on it with a complexity of O ($2^{K/2}$). In this way a cipher having 80 bit key can be attacked with a complexity of order O(2^{40}) and this complexity can be easily achieved with the recent advancement in hardware technology. Hence it was needed that the minimum of stream cipher key should now be assumed as 128 bits. This was the motive behind the new 128 bit version of Grain called Grain 128 while maintaining the benefits of Grain V_1.

Grain 128 uses a 128 bit LFSR and a 128 bit NFSR that provides a 256 bit internal state equally divided among LFSR and NFSR while other design principles remained same. The Boolean function h(x) was also modified.

The feedback polynomials and update functions of LFSR and NFSR were updated accordingly.

Feedback polynomial of LFSR

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

It is a irreducible primitive polynomial of degree 128.

The update function of LFSR is defined as:

$$s_{i+128} = s_i + s_{i+7} + s_{i+38} + s_{i+70} + s_{i+81} + s_{i+96}$$

The feedback polynomial of NFSR is defined as:

$$g(x) = 1 + x^{32} + x^{37} + x^{72} + x^{102} + x^{128} + x^{44}x^{60} + x^{61}x^{125} + x^{63}x^{67} + x^{69}x^{101} + x^{80}x^{88} + x^{110}x^{111} + x^{115}x^{117}$$

Now the update function of NFSR is defined as:

$$b_{i+128} = s_i + b_i + b_{i+26} + b_{i+56} + b_{i+91} + b_{i+96} + b_{i+3}b_{i+67} + b_{i+11}b_{i+13} + b_{i+17}b_{i+18} + b_{i+27}b_{i+59} + b_{i+40}b_{i+48} + b_{i+61}b_{i+65} + b_{i+68}b_{i+84}$$

The filter function is defined as:

$$h(x) = x_0x_1 + x_2x_3 + x_4x_5 + x_6x_7 + x_0x_4x_8$$

Where two inputs are taken from NFSR and seven inputs from LFSR and the variables x_0 to x_8 respectively correspond to the tap position b_{i+12}, s_{i+8}, s_{i+13}, s_{i+20}, b_{i+95}, s_{i+42}, s_{i+60}, s_{i+79} and s_{i+95}.

The keystream function is defined as :

$$z_i = \sum_{j\in A}^{n} b_{i+j} + h(x) + s_{i+93}$$

Where A = {2,15, 36, 45, 64, 73, 89}

Attacks on Grain 128:

Due to similarity in the designs of Grain V_1 and Grain 128, the attacks that are applicable to Grain V1 are also applicable to Grain 128. The attack Proposed by Lee et al [14] takes $2^{26.59}$ chosen IVs, $2^{31.39}$ keystream bits and $2^{27.01}$ computations to recover the 128 bit key.

Berzati et al [17] introduced a fault attack against Grain 128 that can calculate 128 bit key within minutes by using an average 24 consecutive faults in LFSR.

Karmakar and Chowdhury [18] also proposed a fault attack against Grain 128 that targets NFSR and requires 56 faults to upto 256 faults in NFSR state to compute the secret key with time a complexity of O (2^{21}) and space complexity of O (2^{22}).

Dynamic Cube attack [19] was proposed against Grain 128 by Dinur and Shamir that can recover the full key in practical time complexity when initialization rounds is reduced to 207 but when initialization rounds are reduced to 250 only then the time complexity is reduced by a factor of 228 in comparison to exhaustive key search attack.

Dinur et al presented a key recovery attack with the help of a dedicated reconfigurable hardware and based on cube testers [20] that can reduce the attack complexity by a factor of 238 in comparison to exhaustive key search attack. The test results have been experimentally verified by the attackers.

D. Grain 128a:

In order to add Message Authentication Code (MAC) functionality and to overcome the weaknesses in the design in the Grain 128, the designers of Grain have proposed a new design called Grain 128a where a represents authentication.

Grain 128a is the strongest member of Grain family of stream cipher that is 128 bit cipher which also incorporate an authentication mechanism. This design uses the same feedback polynomial for LFSR and similar filter function as in the Grain 128 but the feedback polynomial has been strengthened in view of different attacks proposed against Grain 128.

The new Feedback polynomial of NFSR

$$\boldsymbol{g(x)} = 1 + x^{32} + x^{37} + x^{72} + x^{102} + x^{128} + x^{44}x^{60} + x^{61}x^{125} + x^{63}x^{67} + x^{69}x^{101} + x^{80}x^{88} + x^{110}x^{111} + x^{115}x^{117} + x^{46}x^{50}x^{58} + x^{103}x^{104}x^{106} + x^{33}x^{35}x^{36}x^{40}$$

Now the update function of NFSR is defined as:

$$\boldsymbol{b_{i+128}} = s_i + b_i + b_{i+26} + b_{i+56} + b_{i+91} + b_{i+96} + b_{i+3}b_{i+67} + b_{i+11}b_{i+13} + b_{i+17}b_{i+18} + b_{i+27}b_{i+59} + b_{i+40}b_{i+48} + b_{i+61}b_{i+65} + b_{i+68}b_{i+84} + b_{i+88}b_{i+92}b_{i+93}b_{i+95} + b_{i+22}b_{i+24}b_{i+25} + b_{i+70}b_{i+78}b_{i+82}$$

The filter function is same as in Grain 128 but the keystream function has been also tweaked for Grain 128a. The keystream function is defined as :

$$y_i = h(x) + s_{i+93} + \sum_{j\in A}^{n} b_{i+j}$$

Where A = {2, 15, 36, 45, 64, 73, 89}

$$z_i = y_{64+2i}$$

Grain 128a can be used in both the modes i.e. with authentication or without authentication.

Attacks on Grain 128a:

In case of Grain 128a, the first 64 bits cannot be accessed by the attackers when authentication mode is on. Banik, Maitra and Sarkar proposed a differential fault attack [21] that targets the MAC instead of keystream. This attack requires 2^{11} fault injections and 2^{12} MAC generation routines to access the key.

A second attack was proposed by Ding and Guan [22]. This related key attack requires 2^{96} chosen IVs and $2^{103.613}$ keystream bits to recover the 128 bit key with the computational complexity of $2^{96.322}$.

VII. Comparative Study of Grain Family of Stream Ciphers

In this section, we have discussed and compared the various design parameters for different members of Grain family of Stream ciphers.

In Table 1, we have given the key length IV size and padding used in IV's to fill it for different ciphers of Grain family.

Table 1: Key and IV length in Grain Family of Ciphers

Cipher	Key Length	IV Length	Padding within IV
Grain V0	80	64	FFFF
Grain V1	80	64	FFFF
Grain 128	128	96	FFFFFFFF
Grain 128a	128	96	FFFFFFFE

Only in the last version of Grain family called Grain 128a, the padding is done by all ones except the rightmost bit of LFSR that is filled with zero to avoid the resynchronization attack proposed by Kucuk [8]. In all other versions of Grain, the padding is done with all ones.

In Table 2, we have given the update functions of all the ciphers of the Grain family for the two shift registers i.e. LFSR and NFSR.

Table 2: Update functions of Grain Family of Ciphers

Cipher	LFSR update function	NFSR update function
Grain V_0	$s_{i+80} = s_i + s_{i+13} + s_{i+23} + s_{i+38} + s_{i+51} + s_{i+62}$	$b_{i+80} = s_i + b_{i+63} + b_{i+60} + b_{i+52} + b_{i+45} + b_{i+37} + b_{i+33} + b_{i+28} + b_{i+21} + b_{i+15} + b_{i+9} + b_i + b_{i+63}b_{i+60} + b_{i+37}b_{i+33} + b_{i+15}b_{i+9} + b_{i+60}b_{i+52}b_{i+45} + b_{i+33}b_{i+28}b_{i+21} + b_{i+63}b_{i+45}b_{i+28}b_{i+9} + b_{i+60}b_{i+52}b_{i+37}b_{i+33} + b_{i+63}b_{i+60}b_{i+21}b_{i+15} + b_{i+63}b_{i+60}b_{i+52}b_{i+45}b_{i+37} + b_{i+33}b_{i+28}b_{i+21}b_{i+15}b_{i+9} + b_{i+52}b_{i+45}b_{i+37}b_{i+33}b_{i+28}b_{i+21}$
Grain V_1	$s_{i+80} = s_i + s_{i+13} + s_{i+23} + s_{i+38} + s_{i+51} + s_{i+62}$	$b_{i+80} = s_i + b_i + b_{i+9} + b_{i+14} + b_{i+21} + b_{i+28} + b_{i+33} + b_{i+37} + b_{i+45} + b_{i+52} + b_{i+60} + b_{i+62} + b_{i+9}b_{i+15} + b_{i+33}b_{i+37} + b_{i+60}b_{i+63} + b_{i+21}b_{i+28}b_{i+33} + b_{i+45}b_{i+52}b_{i+60} + b_{i+15}b_{i+21}b_{i+60}b_{i+63} + b_{i+33}b_{i+37}b_{i+52}b_{i+60} + b_{i+9}b_{i+28}b_{i+45}b_{i+63} + b_{i+9}b_{i+15}b_{i+21}b_{i+28}b_{i+33} + b_{i+37}b_{i+45}b_{i+52}b_{i+60}b_{i+63} + b_{i+21}b_{i+28}b_{i+33}b_{i+37}b_{i+45}b_{i+52}$
Grain 128	$s_{i+128} = s_i + s_{i+7} + s_{i+38} + s_{i+70} + s_{i+81} + s_{i+96}$	$b_{i+128} = s_i + b_i + b_{i+26} + b_{i+56} + b_{i+91} + b_{i+96} + b_{i+3}b_{i+67} + b_{i+11}b_{i+13} + b_{i+17}b_{i+18} + b_{i+27}b_{i+59} + b_{i+40}b_{i+48} + b_{i+61}b_{i+65} + b_{i+68}b_{i+84}$
Grain 128a		$b_{i+128} = s_i + b_i + b_{i+26} + b_{i+56} + b_{i+91} + b_{i+96} + b_{i+3}b_{i+67} + b_{i+11}b_{i+13} + b_{i+17}b_{i+18} + b_{i+27}b_{i+59} + b_{i+40}b_{i+48} + b_{i+61}b_{i+65} + b_{i+68}b_{i+84} + b_{i+88}b_{i+92}b_{i+93}b_{i+95} + b_{i+22}b_{i+24}b_{i+25} + b_{i+70}b_{i+78}b_{i+82}$

In table 3, we have given the gate count of different members of the Grain family of ciphers that reflect the hardware complexity of the design.

Table 3: Gate Count for hardware implementation of Grain Family of Ciphers

Cipher	Gate Count for LFSR	Gate Count for NFSR	Gate Count for output function	Total Gate Count
Grain V0	640	640	na	1435
Grain V1	640	640	na	1450
Grain 128	1024	1024	35.5	2133
Grain 128a without authentication	1024	1024	35.5	2145.5
Grain 128a with authentication	1024	1024	35.5	2769.5

As the design of Grain V0 and Grain V1 are similar, hence total gate count is very much equal. Grain 128a without authentication requires just 12.5 gate counts more than Grain 128 that means that Grain 128a can be efficiently used without authentication with comparable hardware complexity of Grain 128 and much more secure than it. Grain 128a with authentication requires just about 30% of more gate counts that means it does not require very much extra hardware for authentication process.

In table 4, we have compared the various members of Grain family of stream ciphers on the basis of key setup time, IV setup time and encryption speed. These encryption speeds have been measured on Pentium 4 2.80 GHz processor machines for two types of data, one for long streams and second for short streams of data less than 40 bytes. Apart from the encryption speed of the all the members of Grain family, the encryption speed of standard block cipher called Advanced Encryption Standard (AES) in counter mode has been also given for comparative purpose. Block cipher in Counter mode of operation (CTR) works as the synchronous stream cipher.

Table 4: Performance comparison of Grain Family of Cipher [23]

Cipher	Key Setup Time	IV Setup Time	Encryption Speed	
			For long streams	For 40 bytes
Grain V_0	29.27	73408.44	3729.79	5545.83
Grain V_1	31.14	1498.23	57.31	102.95
Grain 128	38.89	1098.61	31.16	70.30
AES-CTR with 128 bit key	393.45	76.16	26.86	38.65

This table shows that AES-CTR is better suited in terms of speed, but due to hardware efficiency of the Grain family of stream ciphers, Grain is preferred over AES counter mode in hardware applications.

VIII. Conclusions

In this paper, we have presented the detailed design specifications of the Grain family of stream ciphers and their features. We have studied the major weakness and different attacks on these stream ciphers. We have also presented a comparative study based on hardware and software performance of Grain family of stream cipher, encryption speed, key and IV setup time, etc. The results show that Grain family of stream ciphers is better suited for hardware based applications but the design have some inherent weaknesses that resulted in many cryptanalytic attacks on the ciphers of this family.

References

[1] Rueppel, Rainer A. Analysis and design of stream ciphers. Springer-Verlag New York, Inc., 1986.

[2] Shamir, A. "Stream Ciphers: Dead or Alive?" invited talk, ASIACRYPT 2004, Jeju Island." Korea, Dec (2004): 5-9.

[3] Babbage, Steve. "Stream ciphers: What does the industry want?" State of the Art of Stream Ciphers workshop, Brugge. 2004.

[4] M. Hell, T. Jonasson, and W. Meier. Grain- A Stream Cipher for Constrained Environments. ECRYPT Stream Cipher Project Report 2005/001, 2005. Available at http://www.ecrypt.eu.org/stream.

[5] Robshaw, Matthew. "The eSTREAM project." New Stream Cipher Designs. Springer Berlin Heidelberg, 2008. 1-6.

[6] Hell, Martin, Thomas Johansson, and Willi Meier. "Grain: a stream cipher for constrained environments." International Journal of Wireless and Mobile Computing 2.1 (2007): 86-93.

[7] Hell, Martin, et al. "A stream cipher proposal: Grain-128." Information Theory, 2006 IEEE International Symposium on. IEEE, 2006.

[8] Agren, Martin, et al. "A new version of Grain-128 with authentication."Symmetric Key Encryption Workshop. 2011.

[9] Hell, Martin, et al. "The Grain family of stream ciphers." New Stream Cipher Designs. Springer Berlin Heidelberg, 2008. 179-190.

[10] Küçük, Ö. "Slide resynchronization attack on the initialization of grain 1.0."eSTREAM, ECRYPT Stream Cipher Project, Report 44 (2006): 2006.

[11] Khazaei, Shahram, Mehdi Hassanzadeh, and Mohammad Kiaei. "Distinguishing attack on grain." 2005-12-01)[2009-01-12]. http://www. ecrypt. eu. org/stream/papersdir/071. Pdf (2005).

[12] Berbain, Côme, Henri Gilbert, and Alexander Maximov. "Cryptanalysis of grain."Fast Software Encryption. Springer Berlin Heidelberg, 2006.

[13] De Cannière, Christophe, Özgül Küçük, and Bart Preneel. "Analysis of Grain's initialization algorithm." Progress in Cryptology–AFRICACRYPT 2008. Springer Berlin Heidelberg, 2008. 276-289.

[14] Lee, Yuseop, et al. "Related-key chosen IV attacks on Grain-v1 and Grain-128."Information Security and Privacy. Springer Berlin Heidelberg, 2008.

[15] T.E. Bjørstad. Cryptanalysis of grain using time / memory /data tradeoffs. Available at http://www.ecrypt.eu.org/stream/papersdir/2008/012.pdf.

[16] Dinur, Itai, and Adi Shamir. "Breaking Grain-128 with dynamic cube attacks."Fast Software Encryption. Springer Berlin Heidelberg, 2011.

[17] Berzati, Alexandre, et al. "Fault analysis of GRAIN-128." Hardware-Oriented Security and Trust, 2009. HOST'09. IEEE International Workshop on. IEEE, 2009.

[18] Karmakar, Sandip, and Dipanwita Roy Chowdhury. "Fault analysis of grain-128 by targeting NFSR." Progress in Cryptology–AFRICACRYPT 2011. Springer Berlin Heidelberg, 2011. 298-315.

[19] Dinur, Itai, and Adi Shamir. "Breaking Grain-128 with

dynamic cube attacks."Fast Software Encryption. Springer Berlin Heidelberg, 2011.
[20] Dinur, Itai, et al. "An experimentally verified attack on full Grain-128 using dedicated reconfigurable hardware." Advances in Cryptology–ASIACRYPT 2011. Springer Berlin Heidelberg, 2011. 327-343.
[21] Banik, Subhadeep, Subhamoy Maitra, and Santanu Sarkar. "A differential fault attack on grain-128a using MACs." Security, Privacy, and Applied Cryptography Engineering. Springer Berlin Heidelberg, 2012. 111-125.
[22] Ding, Lin, and Jie Guan. "Related Key Chosen IV Attack on Grain-128a Stream Cipher." Information Forensics and Security, IEEE Transactions on 8.5 (2013): 803-809.
[23] De Canniere, Christophe. "eSTREAM Software Performance." New Stream Cipher Designs. Springer Berlin Heidelberg, 2008. 119-139. Available at http://www.ecrypt.eu.org/stream/phase3perf/2007a/pentium-4-a/ accessed 19/12/2013.

Mohammad Ubaidullah Bokhari, born in 1979. He is currently working as Associate Professor and Ex-Chairman, Department of Computer Science, AMU, Aligarh and has more than 24 years of teaching and research experience. He completed his Ph.D. in Computer Science from AMU, Aligarh. He has published more than 85 research papers in different reputed journals and conference proceedings. He has also authored 5 books on different fields of Computer Science. His current research interests are Cryptography Requirement Engineering, Software Reliability, Wireless Network Security and Database.

Shadab Alam, born in 1985. He is a Ph.D. candidate at Aligarh Muslim University, Aligarh and received his B.Sc. and MCA degrees from Aligarh Muslim University, Aligarh, India. He is pursuing Ph.D. in the field of Cryptography from AMU, Aligarh. He is also working as a counselor for IGNOU. He has published 10 research papers in different reputed international/national journals and conference proceedings. His main research interests include Stream Ciphers, Network Security and Cryptographic Primitives.

Syed Hamid Hasan, has completed his Ph.D. in Computer Science from JMI, India, MSc in Statistics and PGDCS from AMU, India. Dr Hamid has a teaching and research experience of more than 30 years and is currently working as a Professor at Information Systems department, faculty of Computing and Information Technology, King Abdulaziz University, Kingdom of Saudi Arabia. Prof. Hamid has worked as the Head of Computer Science department at AMU, India and also Head of IT department at the Musana College of Technology, Sultanate of Oman.

How to cite this paper: Mohammad Ubaidullah Bokhar, Shadab Alam, Syed Hamid Hasan,"A Detailed Analysis of Grain family of Stream Ciphers", IJCNIS, vol.6, no.6, pp.34-40, 2014. DOI: 10.5815/ijcnis.2014.06.05

