Advances in Intelligent Systems Research (AISR), volume 142
2017 International Conference on Information Technology and Intelligent Manufacturing (ITIM 2017)

Security Evolution on the Non-linear Part of SNOW2.0 against Guess and Determine Attack

Hao Hu [1], Jianjun Lu [2]
[1] The 27th Research Institute of China Electronics Technology Group Corporation
Zhengzhou, China
[2] Air Force Engineering University
Xinyang, China
Email: huhao27@163.com

Abstract. ***SNOW 2.0 was proposed by Ekdahl and Johansson as a strengthened version of SNOW 1.0, which was submitted to the NESSIE project, with a variable-length key of 256 bits. The designers of SNOW2.0 improved the resistance against Guess and Determine (GD) attack by introducing two inputs to the Finite State Machine (FSM). In this paper, the results show that the introduction of those two inputs is not optimal. The suggestion on improving the resistance against GD attack for SNOW2.0 is given.***

Keywords-SNOW2.0; Guess and Determine Attack; SNOW1.0; Finite State Machine

1 Introduction

The original version denoted SNOW1.0[1] was submitted to the NESSIE project. It has excellent performance, several faster than AES. However, a few attacks have been reported. SNOW2.0 was proposed by Ekdahl and Johansson in [2] as a strengthened version of SNOW1.0. Currently, SNOW2.0 is considered as one of the most efficient stream ciphers. It is used for benchmarking the performance of stream ciphers by the eSTREAM project. SNOW2.0 has also been taken as a starting point for the ETSI project on a design of a new UMTS encryption algorithm [3]. Because of efficient implementation both in hardware and software, SNOW 2.0 is one out of two stream ciphers chosen for the forthcoming international standard ISO/IEC IS 18033-4[3].

Guess and Determine (GD) attack can be considered as one of the general attacks on stream ciphers. Arising from the name, in GD attacks, the contents of some cells are to be guessed, based on which the contents of the other cells of the stream cipher can be determined. In [4] a systematic way of implementing some GD attacks by solving systems of linear equations, called Advanced GD attacks, is introduced. The result of implementing Advanced GD attacks on SNOW 2.0 shows the complexity of $O\left(2^{267}\right)$, while there is no heuristic GD attack introduced on SNOW 2.0. In [5], it has been shown that there is a linear distinguisher on SNOW 2.0 which requires 2^{177} bits of keystream and 2^{172} operations. In 2008, Jung-Keun Lee et al.[6] presented a correlation attack on SNOW2.0 with a computational complexity of $O\left(2^{212.38}\right)$, a Memory complexity of $O\left(2^{202.83}\right)$ bits, a data complexity of $O\left(2^{198.77}\right)$ bits.

The main changes from SNOW 1.0 to SNOW 2.0 were to modify the feedback polynomial and to ensure that the FSM takes two inputs from the shift register. The introduction of two inputs to the FSM part makes a guess-and-determine attack more difficult. But the designers of SNOW2.0 did not show the method of introducing these two inputs in [2]. In this paper, we will show the optimal inputs introduced to the FSM part to improve the resistance against GD attacks.

In section 2 a short description of SNOW 2.0 is given. In section 3 we show the optimal inputs introduced to the FSM part. We give an overall view on the paper along with suggestions on improving the resistance against GD attacks for SNOW 2.0 in section 4.

2 a short description of SNOW 2.0

SNOW2.0 is a word-oriented stream cipher with 16-word internal state. Each word consists of 32 bits. The keystream generation of SNOW2.0 can be grouped under roughly 3 parts: Linear Feedback Shift Register (LFSR), Finite State Machine (FSM), and output Transformation. The bitwise XOR of two 32-bit blocks is denoted by $\oplus$ and addition modulo 2^{32} is denoted by $\boxplus$.

Figure.1 is a schematic picture of SNOW2.0.

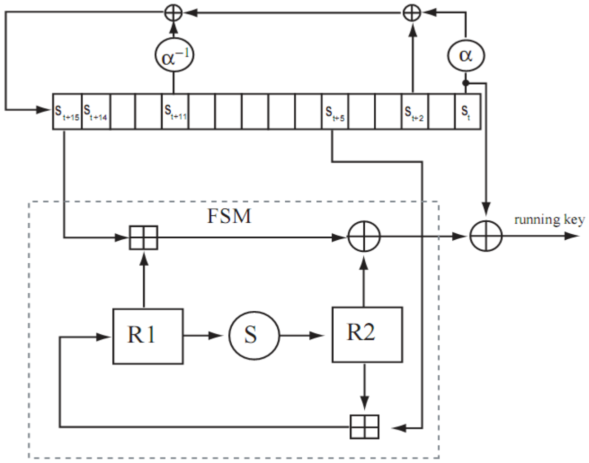

Fig 1. A schematic picture of SNOW2.0

In SNOW2.0, we have two different elements involved in the feedback loop, α and α^{-1}, where α is a root of primitive polynomial of degree 4 over F_{2^8}. To be more precise, the LFSR consists of sixteen 32-bit registers and is associated with the feedback polynomial over GF(2^{32}) as follows.

$$\pi\left(x\right) = \alpha x^{16} + x^{14} + \alpha^{-1}x^{5} + 1 \in F_{2^{32}}\left[x\right] \qquad (1)$$

Here, α is a root of $X^4 + \beta^{23}X^3 + \beta^{245}X^2 + \beta^{48}X + \beta^{239}$ $\in F_{2^8}\left[x\right]$,and β is a root of $X^8+X^7+X^5+X^3+1$ $\in F_2\left[x\right]$.

Let the state of LFSR at time $t \geq 0$ be denoted $\left(s_{t+15},\cdots,s_t\right), s_{t+i} \in F_{2^{32}}\left[x\right], i \geq 0.$ So the recursive relationship between the LFSR and states is as follows.

$$s_{t+16} = s_{t+2} + \alpha^{-1}s_{t+11} + \alpha s_t \qquad (2)$$

Thus, by squaring (2) we have

$$s_{t+32} = s_{t+4} + \alpha^{-2}s_{t+22} + \alpha^{2}s_t \qquad (3)$$

The FSM has two registers, denoted $R1$ and $R2$, each holding 32 bits. The value of the registers at time $t \geq 0$ is denoted $R1_t$ and $R2_t$, respectively. The input to the FSM is $\left(s_{t+15}, s_{t+5}\right)$ and the output of the FSM, denoted f_t, is calculated as

$$f_t = (s_{t+15} \boxplus R1_t) \oplus R2_t \qquad (4)$$

Then the output z_t of the keystream generator is given as

$$z_t = f_t \oplus s_t \qquad (5)$$

The FSM is updated as follows.

$$R1_{t+1} = s_{t+5} \boxplus R2_t \qquad (6)$$

$$R2_{t+1} = S\left(R1_t\right) \qquad (7)$$

The S-box, denoted S, is a permutation on $\mathbb{Z}_{2^{32}}$ based on the round function of Rijndael [7].

3 The analysis of optimal inputs introduced to the FSM part

In [2], the authors claim that the FSM taking two inputs making GD attacks more difficult. Because given the output of FSM, together with $R1$ and $R2$ is no longer possible to deduce the next FSM state directly. The update of $R1$ does not depend on the output of the FSM, but on a word taken from the LFSR. Hence, the introduction of s_{t+5} in relation (6) result in improving the resistance against GD attack and correlation attack. But the authors do not show the reason of introducing s_{t+5} not other states of LFSR.

In fact, the introduction of s_{t+5} in relation (6) is not optimal for improving the resistance against GD attack according to our research. Here, we will replace s_{t+5} in relation (6) with other states of LFSR, keeping other parts of SNOW2.0 unchanged. Then we give GD attacks on each kind of modified SNOW2.0. For comparison we also give a GD attack on unmodified SNOW2.0. The results are depicted in Table 1. In this table, the inputs denote the states of LFSR introduced to the FSM part.

Table 1. The results of GD attacks on each kind of modified SNOW2.0

The input	The elements guessed for GD attack	Computational complexity
s_t	$R1_t, R2_t, s_{t-1}, s_t, s_{t+1}, s_{t+4}, R1_{t+3}, R1_{t+4}$	$O\left(2^{256}\right)$
s_{t+1}	$R1_t, R2_t, s_{t-1}, s_t, s_{t+1}, s_{t+2}, s_{t+4}, R1_{t+3}$	$O\left(2^{256}\right)$
s_{t+2}	$R1_t, R2_t, s_{t-1}, s_t, s_{t+1}, s_{t+2}, s_{t+3}, R1_{t+3}$	$O\left(2^{256}\right)$
s_{t+3}	$R1_t, R2_t, s_{t-1}, s_t, s_{t+2}, s_{t+3}, s_{t+4}, R1_{t+3}, R1_{t+6}$	$O\left(2^{288}\right)$
s_{t+4}	$R1_t, R2_t, s_{t-1}, s_t, s_{t+3}, s_{t+4}, s_{t+5}, R1_{t+3}, R1_{t+6}$	$O\left(2^{288}\right)$
s_{t+5}	$R1_t, R2_t, s_t, s_{t+1}, s_{t+4}, s_{t+5}, s_{t+6}, R1_{t+3}, R1_{t+5}$	$O\left(2^{288}\right)$
s_{t+6}	$R1_t, R2_t, s_t, s_{t+1}, s_{t+5}, s_{t+6}, s_{t+7}, R1_{t+3}, R1_{t+5}$	$O\left(2^{288}\right)$
s_{t+7}	$R1_t, R2_t, s_{t+1}, s_{t+4}, s_{t+5}, s_{t+6}, s_{t+7}, s_{t+8}, R1_{t+3}, R1_{t+5}$	$O\left(2^{320}\right)$
s_{t+8}	$R1_t, R2_t, s_t, s_{t+1}, s_{t+2}, s_{t+7}, s_{t+8}, s_{t+9}, R1_{t+3}, R1_{t+5}$	$O\left(2^{320}\right)$
s_{t+9}	$R1_t, R2_t, s_{t+1}, s_{t+2}, s_{t+3}, s_{t+4}, s_{t+8}, s_{t+9}, s_{t+10}, R1_{t+3}$	$O\left(2^{320}\right)$
s_{t+10}	$R1_t, R2_t, s_{t+2}, s_{t+3}, s_{t+4}, s_{t+9}, s_{t+10}, s_{t+11}, R1_{t+3}$	$O\left(2^{288}\right)$
s_{t+11}	$R1_t, R2_t, s_{t+2}, s_{t+3}, s_{t+10}, s_{t+11}, s_{t+12}, R1_{t+3}$	$O\left(2^{256}\right)$
s_{t+12}	$R1_t, R2_t, s_{t+3}, s_{t+4}, s_{t+11}, s_{t+12}, s_{t+13}, R1_{t+3}$	$O\left(2^{256}\right)$
s_{t+13}	$R1_t, R2_t, s_{t+10}, s_{t+11}, s_{t+12}, s_{t+13}, s_{t+14}, R1_{t+3}$	$O\left(2^{256}\right)$
s_{t+14}	$R1_t, R2_t, s_{t+2}, s_{t+3}, s_{t+13}, s_{t+14}, s_{t+15}, R1_{t+3}$	$O\left(2^{256}\right)$
s_{t+15}	$R1_t, R2_t, s_{t+14}, s_{t+15}, s_{t+16}, R1_{t+3}, R1_{t+4}, R1_{t+5}$	$O\left(2^{256}\right)$

According to table 1, we can see that the input s_{t+5} introduced to the FSM part is not optimal for improving the resistance against GD attack as the designers claimed. The results in table 1 show that replacing s_{t+5} with one of three states $s_{t+7}, s_{t+8}, s_{t+9}$ can improve the resistance against GD attack for SNOW 2.0. This is our suggestion on modifying SNOW2.0. As for Advanced GD attack on SNOW2.0 introduced in [4], which is through solving systems of linear equations, it is mainly based on the linear relations (2) and (3). Here we keep these two relations unchanged, only modifying the input s_{t+5} introduced to the FSM part. Hence, we think that the complexity of Advanced GD attack on modified SNOW2.0 will keep about $O\left(2^{267}\right)$ unchanged.

4. Conclusions

The designers of SNOW2.0 improved the resistance against Guess and Determine (GD) attack by introducing two inputs to the Finite State Machine (FSM). In this paper, the results show that the introduction of those two inputs is not optimal. The suggestion on improving the resistance against GD attack for SNOW2.0 is given.

References

[1] P.Ekdahl,T.Johansson.SNOW–a new stream cipher.Proceedings of first NESSIE workshop,Heverlee,Belgium,2000.
[2] P. Ekdahl and T. Johansson. “A New Version of the Stream Cipher SNOW”, Selected Areas in Cryptography, SAC 2002, LNCS 2295, pp.47-61, Springer Verlag, 2002.
[3] ETSI/SAGE.Specification of the 3GPP confidentiality and integrity algorithms UEA2&UIA2.Document 5:Design and evaluationreport,version:1.0,2006.http://www.3gpp.org/

ftp/tsg_sa/WG3_Security/TSGS3_42_Bangalore/Docs/S3060 180.zip

[4] H. Ahmadi and T. Eghlidos.“Advanced Guess and Determine Attacks on Stream Cipher”.IST 2005, pp.87-91, 2005.

[5] Nyberg Kaisa and Wallen Johan.Improved Linear Distinguishers for SNOW2.0.International Association of Cryptologic Research,vol 4047,p144-162,springer-Verlag.

[6] Jung-Keun Lee, Dong Hoon Lee, and Sangwoo Park. “Cryptanalysis of Sosemanuk and SNOW2.0 Using Linear Masks”. ASIACRYPT 2008, LNCS 5350, pp.524–538, 2008.

[7] Daemen J, Rijmen V. The design of Rilndael: AES-the Advanced Encryption Standard. Springer-Verlag, Berlin,2002.

[8] P. Hawkes,G. Rose. “Guess-and-determine attacks on SNOW”, Preproceedings of Selected Areas in Cryptography (SAC), August 2002, St John’s, Newfoun