sciendo

TATRA MOUNTAINS
Mathematical Publications
DOI: 10.2478/tmmp-2024-0009
Tatra Mt. Math. Publ. **87** (2024), 55–66

CRYPTANALYSIS OF STRUMOK CIPHER INITIALIZATION

Andrii Fesenko

Institute of Physics and Technology, National Technical University of Ukraine
"Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, UKRAINE

ABSTRACT. The initialization of the *Strumok* cipher is analyzed in this paper. A modified cipher initialization procedure is constructed for the *Strumok-512* cipher, which allows to show that the cipher initialization is not injective. Security against slide attacks is proved. Furthermore, the Guess-and-Determine attack on the *Strumok-512* cipher is constructed with complexity 2^{448}.

1. Introduction

Strumok is a word-oriented stream cipher that was introduced in 2016 in [7]. It completely repeats the structure of the *SNOW 2.0* stream cipher, but is oriented to work with 64-bit words. In addition, modifications have been made to the linear feedback shift register (or LFSR) and finite state machine used. However, in [7] the initialization of the cipher was not described and it was suggested to use 512-bit and 1024-bit keys and 256-bit and 512-bit initialization vectors, respectively. Later papers introduced some changes to the structure of the *Strumok* cipher, especially to the key size and the size of the initialization vector, e.g., in [3]. The *Strumok* cipher was adopted as the national encryption standard of Ukraine in 2019 (standard DSTU 8845 : 2019 [1]). We will refer to the *Strumok* cipher as the cipher described in the DSTU 8845 : 2019 standard [1] to avoid discrepancies with other descriptions.

© 2024 Mathematical Institute, Slovak Academy of Sciences.
2020 Mathematics Subject Classification: 94A60.
Keywords: Strumok cipher, Guess-and-Determine attack, initialization attack.
Licensed under the Creative Commons BY-NC-ND 4.0 International Public License.

There are very few works with security analysis of the *Strumok* cipher. Mostly the speed of the cipher is evaluated as in [2] and the security against related key attacks and correlation attacks was considered in [6]. Thus, the cipher needs a thorough security analysis against various attacks because the DSTU 8845 : 2019 standard contains some additional properties of the *Strumok* cipher that were not stated for other ciphers from the *SNOW* family, such as increased security level and post-quantum security.

2. The Strumok stream cipher

The *Strumok* cipher is completely based on the structure of the *SNOW 2.0* stream cipher, but focuses on working with 64-bit words to provide better performance on the modern computers. Like any other stream cipher from the *SNOW* family the *Strumok* cipher consists of a linear feedback shift register of length 16 over $\mathbb{F}_{2^{64}}$, feeding a finite state machine with two 64-bit registers: r_1 and r_2 (the structure is shown in Figure 1).

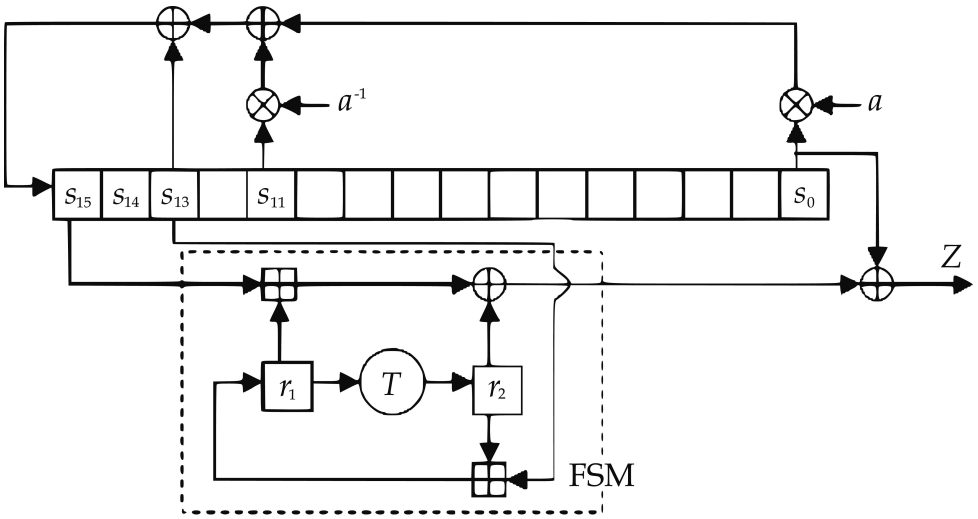


FIGURE 1. The structure of the *Strumok* cipher.

So the main differences in the *Strumok* cipher compared to the *SNOW 2.0* stream cipher are: use of 64-bit words, different LFSR, non-linear transformation is based on the *Kalyna* cipher (from another Ukrainian standard — DSTU 7624:2014) and some changes in the cipher initialization.

The LFSR has a primitive feedback polynomial over $\mathbb{F}_{2^{64}}$, which is

$$f(x) = x^{16} + x^{13} + \alpha^{-1}x^{11} + \alpha,$$

where α is the root of the primitive polynomial over $\mathbb{F}_{2^8}$

$$g(z) = z^8 + \beta^{170}z^7 + \beta^{166}z^6 + \beta^2 z^5 + \beta^{224}z^4 + \beta^{70}z^3 + \beta^2.$$

In turn, $\mathbb{F}_{2^8}$ is generated by the irreducible polynomial over $\mathbb{F}_2$

$$p(y) = y^8 + y^4 + y^3 + y^2 + 1,$$

By design, the *Strumok* cipher algorithm distinguishes three main functions: the internal state initialization function *Init*, which takes as input the secret key K (of either 256 bit or 512 bit length) and the initialization vector IV (known to the public of 256 bit length), and computes the initial value of the state variable $S_0 = (s^{(0)}, r^{(0)})$; state update function taking as input the state value $S_i = (s^{(i)}, r^{(i)})$ and calculating the next value of the state $S_{i+1} = (s^{(i+1)}, r^{(i+1)})$, and which can be executed in two modes — normal and initialization mode; an output function that takes the input state $S_i = (s^{(i)}, r^{(i)})$ and computes a 64-bit value of the key stream Z_i.

It is more convenient to use a slightly different notation and algebraic cipher model to analyze the *Strumok* cipher. To denote elements of the set $\{0,1\}^{64}$ we use lowercase letters, possibly with indices, and to denote strings resulting from concatenation of elements of the set $\{0,1\}^{64}$ we use uppercase letters with the additional notation $\widehat{\ }$, possibly with indices. If $\widehat{X} \in \{0,1\}^{64\cdot n}$ is the result of concatenating the ordered set $x_{n-1}, \ldots, x_0 \in \{0,1\}^{64}$, then we denote this as $\widehat{X} = x_{n-1}||\ldots||x_0$ or as $\widehat{X} = (x_{n-1}, \ldots, x_0)$. The numbering of the components of a line usually starts with the index 0, which indicates the rightmost subword.

For the values of the LFSR cells we will use the notation $s_{15}, \ldots, s_0 \in \{0,1\}^{64}$; for the values of the finite state machine registers — $r_2, r_1 \in \{0,1\}^{64}$; to indicate the internal state of the cipher — $\widehat{W} \in \{0,1\}^{1152}$, where $\widehat{W} = s_{15}||\ldots||s_0||r_2||r_1$, $s_{15}, \ldots, s_0, r_2, r_1 \in \{0,1\}^{64}$; to indicate the element of the key stream — $z \in \{0,1\}^{64}$, if necessary, using the values of the corresponding clock cycle in the superscript for all entered notations.

For elements of the set $\{0,1\}^{64}$ we denote by $\oplus$ the binary addition operation modulo 2; by $\neg$ — unary bitwise negation operation; by $+_{64}$ — binary addition operation modulo 2^{64} (to reduce the notation for this operation, we will also use the notation $+$). By $\mathbf{0}$ we denote the element of the set $\{0,1\}^{64}$ consisting of 64 characters 0, and by $\mathbf{1}$ we denote an element consisting of 64 characters 1.

The mode of the *Strumok* cipher with a 256 bit key is called the *Strumok-256* cipher, and the mode with a 512 bit key is called the *Strumok-512* cipher.

To use an algebraic model of key initialization we need to define additional mappings that use the values of the secret key and the initialization vector to initialize the internal state of the cipher.

Let the mapping

$$\mathrm{SetKey}_{256}\colon \{0,1\}^{256} \times \{0,1\}^{256} \to \{0,1\}^{1152}$$

be defined as follows so that for arbitrary values $\widehat{K} \in \{0,1\}^{256}$ and $\widehat{IV} \in \{0,1\}^{256}$, $\widehat{K} = k_3||\ldots||k_0$, $\widehat{IV} = iv_3||iv_2||iv_1||iv_0$, where $k_3,\ldots,k_0,iv_3,iv_2,iv_1,iv_0 \in \{0,1\}^{64}$, the mapping result is

$$\mathrm{SetKey}_{256}(\widehat{K},\widehat{IV}) = \widehat{W} = s_{15}||\ldots||s_0||r_2||r_1,$$

where

$$\begin{array}{lllll} s_{15} = \neg k_0, & s_{11} = k_0, & s_7 = \neg k_0, & s_3 = k_0 \oplus iv_2, & r_1 = \mathbf{0}, \\ s_{14} = k_1, & s_{10} = \neg k_1, & s_6 = \neg k_1, & s_2 = k_1 \oplus iv_1, & r_2 = \mathbf{0}. \\ s_{13} = \neg k_2, & s_9 = k_2, & s_5 = k_2 \oplus iv_3, & s_1 = k_2, & \\ s_{12} = k_3, & s_8 = k_3, & s_4 = k_3, & s_0 = k_3 \oplus iv_0. & \end{array}$$

Accordingly, the mapping

$$\mathrm{SetKey}_{512}\colon \{0,1\}^{512} \times \{0,1\}^{256} \to \{0,1\}^{1152}$$

is defined as follows so that for arbitrary values $\widehat{K} \in \{0,1\}^{512}$ and $\widehat{IV} \in \{0,1\}^{256}$, $\widehat{K} = k_7||\ldots||k_0$, $\widehat{IV} = iv_3||iv_2||iv_1||iv_0$, where $k_7,\ldots,k_0,iv_3,iv_2,iv_1,iv_0 \in \{0,1\}^{64}$, the mapping result is

$$\mathrm{SetKey}_{512}(\widehat{K},\widehat{IV}) = \widehat{W} = s_{15}||\ldots||s_0||r_2||r_1,$$

where

$$\begin{array}{lllll} s_0 = k_7 \oplus iv_0, & s_4 = k_3, & s_8 = k_4 \oplus iv_3, & s_{12} = k_3, & r_1 = \mathbf{0}, \\ s_1 = k_6, & s_5 = k_2 \oplus iv_2, & s_9 = \neg k_6, & s_{13} = k_2, & r_2 = \mathbf{0}. \\ s_2 = k_5, & s_6 = k_1, & s_{10} = k_5, & s_{14} = \neg k_1, & \\ s_3 = k_4 \oplus iv_1, & s_7 = \neg k_0, & s_{11} = \neg k_7, & s_{15} = k_0, & \end{array}$$

Let the mapping

$$\mathrm{Next}\colon \{0,1\}^{1152} \to \{0,1\}^{1152}$$

be defined so that for an arbitrary value of $\widehat{W} \in \{0,1\}^{1152}$, $\widehat{W} = s_{15}||\ldots||s_0||r_2||r_1$, where $s_{15},\ldots,s_0,r_2,r_1 \in \{0,1\}^{64}$, the value $\mathrm{Next}(\widehat{W}) = s_{16}||s_{15}||\ldots||s_1||\tilde{r}_2||\tilde{r}_1$, where $s_{16},\tilde{r}_2,\tilde{r}_1 \in \{0,1\}^{64}$, $s_{16} = s_{13} \oplus (s_{11} \otimes \alpha^{-1}) \oplus (s_0 \otimes \alpha)$, $\tilde{r}_2 = T(r_1)$, $\tilde{r}_1 = r_2 + s_{13}$.

The mapping

$$\mathrm{Next}_{\mathrm{init}}\colon \{0,1\}^{1152} \to \{0,1\}^{1152}$$

is defined so that for an arbitrary value of $\widehat{W} \in \{0,1\}^{1152}$, $\widehat{W} = s_{15}||\ldots||s_0||r_2||r_1$, where $s_{15},\ldots,s_0,r_2,r_1 \in \{0,1\}^{64}$, the value $\mathrm{Next}(\widehat{W}) = s_{16}||s_{15}||\ldots||s_1||\tilde{r}_2||\tilde{r}_1$, where $s_{16},\tilde{r}_2,\tilde{r}_1 \in \{0,1\}^{64}$, $s_{16} = s_{13} \oplus (s_{11} \otimes \alpha^{-1}) \oplus (s_0 \otimes \alpha) \oplus r_2 \oplus (r_1 + s_{15})$, $\tilde{r}_2 = T(r_1)$, $\tilde{r}_1 = r_2 + s_{13}$.

The mapping

$$\mathrm{Strm}\colon \{0,1\}^{1152} \to \{0,1\}^{64}$$

is defined so that for an arbitrary value of $\widehat{W} \in \{0,1\}^{1152}$, $\widehat{W} = s_{15}||\ldots||s_0||r_2||r_1$, where $s_{15},\ldots,s_0,r_2,r_1 \in \{0,1\}^{64}$, the value $\mathrm{Strm}(\widehat{W}) = s_0 \oplus r_2 \oplus (s_{15} + r_1)$.

For any fixed key value $\widehat{K} \in \{0,1\}^{256}$ and initialization vector $\widehat{IV} \in \{0,1\}^{256}$, the internal state of the cipher *Strumok-256* at time (-33) is $W^{(-33)} = \mathrm{SetKey}_{256}(\widehat{K}, \widehat{IV})$.

For moments of time $t \in \{-32, \ldots, -1\}$ the internal state of the cipher *Strumok-256* at time t is determined iteratively $W^{(t)} = \mathrm{Next}_{\mathrm{init}}(W^{(t-1)})$. The internal state $W^{(0)}$ of the cipher *Strumok-256* at the time 0 is defined as $W^{(0)} = \mathrm{Next}(W^{(-1)})$. For the time moments $t \in \mathbb{N}$, $1 \le t \le 2^{80}$, the internal state of the cipher *Strumok-256* at the time t is determined iteratively $W^{(t)} = \mathrm{Next}(W^{(t-1)})$, and the corresponding value of the key stream at the time t, $t \in \mathbb{N}$, is defined as $z^{(t)} = \mathrm{Strm}(W^{(t)})$.

Similarly, for arbitrary fixed values of the key $\widehat{K} \in \{0,1\}^{512}$ and the initialization vector $\widehat{IV} \in \{0,1\}^{256}$, the internal state of the cipher *Strumok-512* at the time (-33) is $W^{(-33)} = \mathrm{SetKey}_{512}(\widehat{K}, \widehat{IV})$. For moments of time $t \in \{-32, \ldots, -1\}$ the internal state of the *Strumok-512* cipher at time t is determined iteratively $W^{(t)} = \mathrm{Next}_{\mathrm{init}}(W^{(t-1)})$. The internal state $W^{(0)}$ of the cipher *Strumok-512* at the time 0 is defined as $W^{(0)} = \mathrm{Next}(W^{(-1)})$. For time moments $t \in \mathbb{N}$, $1 \le t \le 2^{80}$, the internal state of the cipher *Strumok-512* at the time t is determined iteratively $W^{(t)} = \mathrm{Next}(W^{(t-1)})$, and the corresponding value of the key stream at time t, $t \in \mathbb{N}$, is defined as $z^{(t)} = \mathrm{Strm}(W^{(t)})$.

3. Analysis of Strumok initialization

We will start the analysis of the *Strumok* initialization with the analysis of the SetKey mapping. Ciphers from the *SNOW* family usually use words from the key without reordering but in both ciphers *Strumok-256* and *Strumok-512* this behavior has been changed.

A new mapping

$$\mathrm{NewSetKey}_{512}\colon \{0,1\}^{512} \times \{0,1\}^{256} \to \{0,1\}^{1152}$$

is defined as follows, so that for arbitrary values

$$\widehat{K} \in \{0,1\}^{512} \quad \text{and} \quad \widehat{IV} \in \{0,1\}^{256},$$

$$\widehat{K} = k_7 || \ldots || k_0, \quad \widehat{IV} = iv_3 \,||\, iv_2 \,||\, iv_1 \,||\, iv_0, \quad k_7, \ldots, k_0, iv_3, iv_2, iv_1, iv_0 \in \{0,1\}^{64}$$

the mapping result is

$$\mathrm{NewSetKey}_{512}(\widehat{K}, \widehat{IV}) = \widehat{W} = s_{15} || \ldots || s_0 || r_2 || r_1,$$

where

$$\begin{aligned}
&s_0 = k_7, && s_5 = k_2, && s_{10} = k_5, && s_{15} = k_0,\\
&s_1 = k_6 \oplus \mathbf{1}, && s_6 = k_1 \oplus \mathbf{1}, && s_{11} = k_4 \oplus iv_1, && r_1 = \mathbf{0},\\
&s_2 = k_5, && s_7 = k_0 \oplus \mathbf{1}, && s_{12} = k_3, && r_2 = \mathbf{0}.\\
&s_3 = k_4 \oplus \mathbf{1}, && s_8 = k_7 \oplus iv_0, && s_{13} = k_2 \oplus iv_2,\\
&s_4 = k_3, && s_9 = k_6, && s_{14} = k_1,
\end{aligned}$$

We will use the *NewStrumok-512* cipher name to denote a new cipher that is *Strumok-512* cipher with the only change being the use of $\mathrm{NewSetKey}_{512}$ mapping instead of SetKey_{512} in the cipher initialization phase.

We also need another mapping for our analysis —

$$\varphi_{512}\colon \{0,1\}^{512} \times \{0,1\}^{256} \to \{0,1\}^{512} \times \{0,1\}^{256},$$

which is defined so that for arbitrary values $\widehat{K} \in \{0,1\}^{512}$ and $\widehat{IV} \in \{0,1\}^{256}$, $\widehat{K} = k_7 || \ldots ||k_0$, $\widehat{IV} = iv_3 || \ldots ||iv_0$, where $k_7, \ldots, k_0, iv_3, \ldots, iv_0 \in \{0,1\}^{64}$, the mapping result is calculated as

$$\varphi_{512}(\widehat{K}, \widehat{IV}) = (\tilde{k}_7 || \ldots ||\tilde{k}_0, \tilde{iv}_3 || \ldots ||\tilde{iv}_0),$$

where

$$\begin{aligned}
&\tilde{k}_0 = k_0, && \tilde{k}_6 = k_6 \oplus \mathbf{1},\\
&\tilde{k}_1 = k_1 \oplus \mathbf{1}, && \tilde{k}_7 = k_7 \oplus iv_0,\\
&\tilde{k}_2 = k_2 \oplus iv_2, && \tilde{iv}_0 = k_4 \oplus k_7 \oplus iv_0 \oplus iv_3,\\
&\tilde{k}_3 = k_3, && \tilde{iv}_1 = k_4 \oplus k_7 \oplus iv_1,\\
&\tilde{k}_4 = k_4 \oplus iv_1 \oplus \mathbf{1}, && \tilde{iv}_2 = iv_2,\\
&\tilde{k}_5 = k_5, && \tilde{iv}_3 = iv_1.
\end{aligned}$$

Claim 3.1. *The mapping*

$$\varphi_{512}\colon \{0,1\}^{512} \times \{0,1\}^{256} \to \{0,1\}^{512} \times \{0,1\}^{256}$$

is a bijection.

Proof. We will prove this claim by constructing the inverse mapping

$$\varphi_{512}^{-1}\colon \{0,1\}^{512} \times \{0,1\}^{256} \to \{0,1\}^{512} \times \{0,1\}^{256}$$

so that for arbitrary values $\widehat{\widetilde{K}} \in \{0,1\}^{512}$ and $\widehat{\widetilde{IV}} \in \{0,1\}^{256}$, $\widehat{\widetilde{K}} = \widetilde{k}_7 || \ldots ||\widetilde{k}_0$, $\widehat{\widetilde{IV}} = \widetilde{iv}_3 || \ldots ||\widetilde{iv}_0$, where $k_7, \ldots, k_0, iv_3, \ldots, iv_0 \in \{0,1\}^{64}$, the mapping result is calculated as

$$\varphi_{512}^{-1}(\widehat{\widetilde{K}}, \widehat{\widetilde{IV}}) = (k_7 || \ldots ||k_0, iv_3 || \ldots ||iv_0),$$

where

$$\begin{aligned}
&k_0 = \widetilde{k}_0, && k_6 = \widetilde{k}_6 \oplus \mathbf{1},\\
&k_1 = \widetilde{k}_1 \oplus \mathbf{1}, && k_7 = \widetilde{k}_4 \oplus \widetilde{iv}_1 \oplus \mathbf{1},\\
&k_2 = \widetilde{k}_2 \oplus \widetilde{iv}_2, && iv_0 = \widetilde{k}_4 \oplus \widetilde{k}_7 \oplus \widetilde{iv}_1 \oplus \mathbf{1},\\
&k_3 = \widetilde{k}_3, && iv_1 = \widetilde{iv}_3,\\
&k_4 = \widetilde{k}_4 \oplus \widetilde{iv}_3 \oplus \mathbf{1}, && iv_2 = \widetilde{iv}_2,\\
&k_5 = \widetilde{k}_5, && iv_3 = \widetilde{k}_4 \oplus \widetilde{k}_7 \oplus \widetilde{iv}_3 \oplus \widetilde{iv}_0 \oplus \mathbf{1}.
\end{aligned}$$

The identity $\big(\varphi_{512}^{-1}\big(\varphi_{512}(\widehat{K}, \widehat{IV})\big) = (\widehat{K}, \widehat{IV})\big)$ is checked by direct substitution of the corresponding values. □

Remark 1. When defining the function φ_{512}, the value $\widetilde{iv}_3 = iv_1$ was chosen to construct a bijective mapping. Instead of the value iv_1, any other value can be used, which, together with the equality $k_4 \oplus iv_1 = \widetilde{k}_4 \oplus \mathbf{1}$, allows obtaining a unique solution with respect to the values of k_4 and iv_1.

Claim 3.2. *For arbitrary values* $\widehat{K} \in \{0,1\}^{512}$ *and* $\widehat{IV} \in \{0,1\}^{256}$ *we have the identity*

$$\mathrm{SetKey}_{512}(\widehat{K}, \widehat{IV}) = \mathrm{NewSetKey}_{512}\big(\varphi_{512}(\widehat{K}, \widehat{IV})\big).$$

P r o o f. Consider arbitrary values

$$\widehat{K} \in \{0,1\}^{512} \text{ and } \widehat{IV} \in \{0,1\}^{256}, \ \widehat{K} = k_7||\ldots||k_0, \ \widehat{IV} = iv_3||\ldots||iv_0,$$

where

$$k_7, \ldots, k_0, iv_3, \ldots, iv_0 \in \{0,1\}^{64}.$$

Let

$$\varphi_{512}(\widehat{K}, \widehat{IV}) = (\widetilde{k}_7||\ldots||\widetilde{k}_0, \widetilde{iv}_3||\ldots||\widetilde{iv}_0).$$

.

The corresponding equalities can be checked directly using the definitions of the mapping φ_{512} and the mapping $\mathrm{NewSetKey}_{512}$:

$$\begin{aligned}
s_0 &= \widetilde{k}_7 = k_7 \oplus iv_0,\\
s_1 &= \widetilde{k}_6 \oplus \mathbf{1} = (k_6 \oplus \mathbf{1}) \oplus \mathbf{1} = k_6,\\
s_2 &= \widetilde{k}_5 = k_5,\\
s_3 &= \widetilde{k}_4 \oplus \mathbf{1} = (k_4 \oplus iv_1 \oplus \mathbf{1}) \oplus \mathbf{1} = k_4 \oplus iv_1,\\
s_4 &= \widetilde{k}_3 = k_3,\\
s_5 &= \widetilde{k}_2 = k_2 \oplus iv_2,\\
s_6 &= \widetilde{k}_1 \oplus \mathbf{1} = (k_1 \oplus \mathbf{1}) \oplus \mathbf{1} = k_1,\\
s_7 &= \widetilde{k}_0 \oplus \mathbf{1} = k_0 \oplus \mathbf{1} = \neg k_0,\\
s_8 &= \widetilde{k}_7 \oplus \widetilde{iv}_0 = (k_7 \oplus iv_0) \oplus (k_4 \oplus k_7 \oplus iv_0 \oplus iv_3) = k_4 \oplus iv_3,\\
s_9 &= \widetilde{k}_6 = k_6 \oplus \mathbf{1} = \neg k_6,\\
s_{10} &= \widetilde{k}_5 = k_5,\\
s_{11} &= \widetilde{k}_4 \oplus \widetilde{iv}_1 = (k_4 \oplus iv_1 \oplus \mathbf{1}) \oplus (k_4 \oplus k_7 \oplus iv_1) = k_7 \oplus \mathbf{1} = \neg k_7,\\
s_{12} &= \widetilde{k}_3 = k_3,\\
s_{13} &= \widetilde{k}_2 \oplus \widetilde{iv}_2 = (k_2 \oplus iv_2) \oplus iv_2 = k_2,\\
s_{14} &= \widetilde{k}_1 = k_1 \oplus \mathbf{1} = \neg k_1,\\
s_{15} &= \widetilde{k}_0 = k_0,\\
r_1 &= \mathbf{0}, r_2 = \mathbf{0}.
\end{aligned}$$

□

Corollary 3.3. *The NewStrumok-512 cipher and the Strumok-512 cipher work the same (with respect to the mapping φ_{512}).*

From the Claim 3.2 it follows that in some cases, especially when analyzing the cipher, one can consider the *NewStrumok-512* cipher instead of the *Strumok-512* cipher, i.e., use a modified initialization of the internal state of the cipher that is more similar to the initialization of the *SNOW 2.0* cipher.

It appears that the authors of the *Strumok-512* cipher made appropriate changes to the cipher initialization to prevent slide attacks that are applicable to the *SNOW 2.0* cipher (e.g., in [5]).

In fact, during our analysis we were able to prove that attacks with a small round offset are not applicable to the cipher *Strumok-512* cipher.

Theorem 3.4. *Slide attacks with the number of rounds $1, 2, 3, 4, 5, 6$ or 7 are not applicable to Strumok-512 and Strumok-256 ciphers.*

Proof. To prove this statement for the *Strumok-512* cipher, it is easier to use the *NewStrumok-512* cipher with a modified function $\mathrm{NewSetKey}_{512}$, although the same result can be obtained using the original version of the *Strumok-512* cipher.

$$
\begin{array}{llll}
s_0^{(-33)} = k_7, & s_5^{(-33)} = k_2, & s_{10}^{(-33)} = k_5, & s_{15}^{(-33)} = k_0, \\
s_1^{(-33)} = k_6 \oplus \mathbf{1}, & s_6^{(-33)} = k_1 \oplus \mathbf{1}, & s_{11}^{(-33)} = k_4 \oplus iv_1, & r_1^{(-33)} = \mathbf{0}, \\
s_2^{(-33)} = k_5, & s_7^{(-33)} = k_0 \oplus \mathbf{1}, & s_{12}^{(-33)} = k_3, & r_2^{(-33)} = \mathbf{0}. \\
s_3^{(-33)} = k_4 \oplus \mathbf{1}, & s_8^{(-33)} = k_7 \oplus iv_0, & s_{13}^{(-33)} = k_2 \oplus iv_2, & \\
s_4^{(-33)} = k_3, & s_9^{(-33)} = k_6, & s_{14}^{(-33)} = k_1, &
\end{array}
$$

To apply the slide attack with r round shift, $r \in \mathbb{N}$, $r \geq 1$, it is necessary that for all values $i \in \{0, \ldots, 15\}$ the values $s_i^{(-33+r)}$ satisfy the conditions of the function $\mathrm{NewSetKey}_{512}$ for some value of the key $\widetilde{\widetilde{K}} \in \{0,1\}^{512}$ and the initialization vector $\widetilde{\widetilde{IV}} \in \{0,1\}^{256}$, $\widetilde{\widetilde{K}} = \widetilde{k}_7 || \ldots || \widetilde{k}_0$, $\widetilde{\widetilde{IV}} = \widetilde{iv}_3 || \ldots || \widetilde{iv}_0$, where $\widetilde{k}_7, \ldots, \widetilde{k}_0$, $\widetilde{iv}_3, \ldots, \widetilde{iv}_0 \in \{0,1\}^{64}$.

If $r = 1$, then $\widetilde{s}_9^{(-33)} = \widetilde{k}_6$ and $\widetilde{s}_1^{(-33)} = \widetilde{k}_6 \oplus \mathbf{1}$, but $\widetilde{s}_9^{(-33)} = s_9^{(-33+1)} = s_{10}^{(-33)} = k_5$ and $\widetilde{s}_1^{(-33)} = s_1^{(-33+1)} = s_2^{(-33)} = k_5$. It follows from the equations $\widetilde{k}_6 = k_5$ and $\widetilde{k}_6 \oplus \mathbf{1} = k_5$ that such values k_5 and $\widetilde{k}_6$ do not exist, and therefore the slide attack with 1 round shift is not applicable.

If $r = 2$, then $\widetilde{s}_4^{(-33)} = \widetilde{k}_3$ and $\widetilde{s}_{12}^{(-33)} = \widetilde{k}_3$, but $\widetilde{s}_4^{(-33)} = s_4^{(-33+2)} = s_6^{(-33)} = k_1 \oplus \mathbf{1}$ and $\widetilde{s}_{12}^{(-33)} = s_{12}^{(-33+2)} = s_{14}^{(-33)} = k_1$. It follows from the equations $\widetilde{k}_3 = k_1 \oplus \mathbf{1}$ and $\widetilde{k}_3 = k_1$ that such values k_1 and $\widetilde{k}_3$ do not exist, and therefore the slide attack with 2 round shift is not applicable.

Similarly, for the values

$$r = 3, \quad r = 4, \quad r = 5, \quad r = 6, \quad \text{and} \quad r = 7,$$

it is sufficient to consider pairs of values:

$$\widetilde{s}_4^{(-33)} \quad and \quad \widetilde{s}_{12}^{(-33)}, \qquad \widetilde{s}_2^{(-33)} \quad and \quad \widetilde{s}_{10}^{(-33)},$$

$$\widetilde{s}_2^{(-33)} \quad and \quad \widetilde{s}_{10}^{(-33)}, \qquad \widetilde{s}_4^{(-33)} \quad and \quad \widetilde{s}_{12}^{(-33)},$$

$$\widetilde{s}_2^{(-33)} \quad and \quad \widetilde{s}_{10}^{(-33)},$$

respectively, to prove that slide attacks are inapplicable with the corresponding number of rounds.

To prove this statement for the *Strumok-256* cipher, it is also necessary to consider pairs of values

$$\widetilde{s}_4^{(-33)} \quad and \quad \widetilde{s}_{12}^{(-33)} \quad \text{or} \quad \widetilde{s}_7^{(-33)} \quad and \quad \widetilde{s}_{15}^{(-33)}. \qquad \square$$

Applying slide attacks with more rounds than 7 seems impractical, since a sufficient number of nonlinear transformations are already used.

However, the consequence of such an initialization phase for the *Strumok-512* cipher is that only three out of the four initialization vector subwords are actually used.

Theorem 3.5. *The effective length of the initialization vector in the Strumok-512 cipher is only* 192 *bits.*

It is easy to see when using the $\mathrm{NewSetKey}_{512}$ function: the value iv_3 is not used at all, only the values iv_0, iv_1 and iv_2 from the initialization vector are used. In the original function SetKey_{512}, the problem is caused by equalities $s_3 = k_4 \oplus iv_1$ and $s_8 = k_4 \oplus iv_3$ (values k_4, iv_1 and iv_3 are never used elsewhere). So for any key $\widehat{K} = k_7||\ldots||k_4||\ldots||k_0$ and initialization vector $\widehat{IV} = iv_3||iv_2||iv_1||iv_0$ the initialization procedure of the *Strumok-512* cipher will have the same result for all keys $k_7||\ldots||k_4 \oplus \Delta||\ldots||k_0$ and initialization vectors $iv_3 \oplus \Delta||iv_2||iv_1 \oplus \Delta||iv_0$ with an arbitrary value of $\Delta \in \{0,1\}^{64}$. This means that the initialization procedure in the *Strumok-512* cipher is not injective and Theorem 3.5 implies the existence of key-related attacks on the *Strumok-512* cipher.

4. Guess-and-Determine attack

Another part of our analysis focuses on the possibility of applying Guess-and--Determine attacks to the original *Strumok* cipher. In Guess-and-Determine attacks, a guess is made about some of the internal state values of the cipher, and using the known values of the original key stream and functional dependencies, the remaining unknown internal state values are calculated. It is necessary to choose as small a data size as possible in order to guess the values and calculate as much data as possible.

The *Strumok* cipher is large enough for such an approach to be successful, but its security is estimated at 512 bits according to [1], unlike other ciphers from the *SNOW* family. We were able to prove that such an attack can be successful due to wrong choices in LFSR construction.

Theorem 4.1. *Suppose that at some time $i \in \mathbb{N}$ in the normal mode of operation (not the initialization mode), that is $i \geq 0$, the internal state of the Strumok-512 cipher is equal to $\widehat{W}^{(i)} \in \{0,1\}^{1152}$, where*

$$\widehat{W}^{(i)} = s_{15}^{(i)} || \ldots ||s_0^{(i)}||r_2^{(i)}||r_1^{(i)}, \quad s_{15}^{(i)}, \ldots, s_0^{(i)}, r_2^{(i)}, r_1^{(i)} \in \{0,1\}^{64}.$$

Knowing the values $s_{15}^{(i)}$, $s_{14}^{(i)}$, $s_{13}^{(i)}$, $s_{12}^{(i)}$, $s_{11}^{(i)}$, $r_2^{(i)}$, $r_1^{(i)}$ and the generated key stream allows to determine the internal state of the Strumok-512 cipher at any time of its functioning.

Proof. Assume that the values $s_{15}^{(i)}, s_{14}^{(i)}, s_{13}^{(i)}, s_{12}^{(i)}, s_{11}^{(i)}, r_2^{(i)}, r_1^{(i)}$ are known at some point $i \in \mathbb{N}$. Using the cipher functions Strm and Next, we can construct an equation system that determines the next three words of the key stream, namely:

$$\begin{cases} z^{(i)} &= \mathrm{Strm}\big(\widehat{W}^{(i)}\big), \\ z^{(i+1)} &= \mathrm{Strm}\Big(\mathrm{Next}\big(\widehat{W}^{(i)}\big)\Big), \\ z^{(i+2)} &= \mathrm{Strm}\Big(\mathrm{Next}\Big(\mathrm{Next}\big(\widehat{W}^{(i)}\big)\Big)\Big). \end{cases}$$

With the known cipher functions Strm and Next we can rewrite this system of equations as

$$\begin{cases} z^{(i)} &= s_0^{(i)} \oplus r_2^{(i)} \oplus \big(s_{15}^{(i)} + r_1^{(i)}\big), \\ z^{(i+1)} &= s_1^{(i)} \oplus T\big(r_1^{(i)}\big) \oplus \Big(\big(s_{11}^{(i)} \otimes \alpha^{-1}\big) \oplus \big(s_0^{(i)} \otimes \alpha\big) \oplus s_{13}^{(i)} + s_{13}^{(i)} + r_2^{(i)}\Big), \\ z^{(i+2)} &= s_2^{(i)} \oplus T\big(r_2^{(i)} + s_{13}^{(i)}\big) \\ &\qquad \oplus \Big(\big(s_{12}^{(i)} \otimes \alpha^{-1}\big) \oplus \big(s_1^{(i)} \otimes \alpha\big) \oplus s_{14}^{(i)} + s_{14}^{(i)} + T\big(r_1^{(i)}\big)\Big). \end{cases}$$

It follows from this system that

$$\begin{cases} s_0^{(i)} &= z^{(i)} \oplus r_2^{(i)} \oplus \left(s_{15}^{(i)} + r_1^{(i)}\right), \\ s_1^{(i)} &= z^{(i+1)} \oplus T\left(r_1^{(i)}\right) \oplus \Big(\big(s_{11}^{(i)} \otimes \alpha^{-1}\big) \oplus \big(s_0^{(i)} \otimes \alpha\big) \oplus s_{13}^{(i)} + s_{13}^{(i)} + r_2^{(i)}\Big), \\ s_2^{(i)} &= z^{(i+2)} \oplus T\Big(r_2^{(i)} + s_{13}^{(i)}\Big) \\ &\qquad \oplus \Big(\big(s_{12}^{(i)} \otimes \alpha^{-1}\big) \oplus \big(s_1^{(i)} \otimes \alpha\big) \oplus s_{14}^{(i)} + s_{14}^{(i)} + T\big(r_1^{(i)}\big)\Big). \end{cases}$$

The right-hand sides of the equations in the obtained system contain only the known values

$$s_{15}^{(i)}, \quad s_{14}^{(i)}, \quad s_{13}^{(i)}, \quad s_{12}^{(i)}, \quad s_{11}^{(i)}, \quad r_2^{(i)}, \quad r_1^{(i)},$$

as well as the known values of the key stream $z^{(i)}$, $z^{(i+1)}$ and $z^{(i+2)}$. In this way, it is possible to uniquely calculate the values of $s_0^{(i)}$, $s_1^{(i)}$ and $s_2^{(i)}$.

Next, from the known values of

$$s_0^{(i)},\quad s_1^{(i)},\quad s_2^{(i)}\quad\text{and}\quad s_{15}^{(i)},\quad s_{14}^{(i)},\quad s_{13}^{(i)},\quad s_{12}^{(i)},\quad s_{11}^{(i)}$$

we can calculate the values

$$s_{16}^{(i)},\quad s_{17}^{(i)},\quad s_{18}^{(i)}$$

(where notation $s_n^{(i)}$ for $n>15$ is used to denote the value $s_{n\ (\mathrm{mod}\ 16)}^{(i+n-n\ (\mathrm{mod}\ 16))}$ and for all $i\geq -33$, $n,k\geq 0$ we have an identity $s_{n+k}^{(i)}=s_n^{(i+k)}$), and using these values and the values of $z^{(i+3)}$, $z^{(i+4)}$, $z^{(i+3)}$, we also get the values of $s_3^{(i)}$, $s_4^{(i)}$, $s_5^{(i)}$.

Continuing further, we will get all the unknown values of the internal state $\widehat{W}^{(i)}$. From any internal state of the *Strumok-512* cipher we can calculate the next and the previous internal states and it means that we can determine the internal state of the *Strumok-512* cipher at any time of its functioning (up to the state with filled key and initialization vector words, if we know when the cipher started its work in the normal mode). □

Remark 2. The method from the proof of Theorem 4.1 also allows to compute the values of

$$s_{15}^{(i)}, s_{14}^{(i)}, s_{13}^{(i)}, s_{12}^{(i)}, s_{11}^{(i)}$$

from the equations and this fact allows us to check the guess about their values. In the case of a successful guess, the entire value of the internal state will be known.

Corollary 4.2. *There is a Guess-and-Determine attack on the Strumok-512 cipher with complexity* 2^{448}*, which allows to compute the complete internal state of the cipher and allows to compute the key if we know how many steps the cipher worked in the normal mode.*

This attack is specific to the *Strumok-512* cipher and is possible due to the very close left positions of the taps and the fact that s_{13}^i is used twice: for LFSR feedback and for updating Finite State Machine values. It violates the statement of the standard DSTU 8845 : 2019 where the complexity of the best attack on the *Strumok-512* cipher is evaluated as 2^{512}.

The authors in [4] propose a byte-based Guess-and-Determine attack against the *SNOW-V* cipher with complexity 2^{406} using seven key stream words. To make this attack, the state variables were split into bytes with some carriers introduced, and a dynamic programming tool was used to help find a good guessing path that requires guessing as few bytes as possible. They improved the heuristic path-search approach to find a proper (not always optimal) guessing basis in a reasonable time and memory.

We used the same method with variable splitting into bytes, but it did not help to find a Guess-and-Determine attack with better complexity than 2^{448}. The *Strumok* cipher has a huge internal state which makes the exhaustive search impossible but a byte-based Guess-and-Determine attack still looks promising. However, some additional improvements to the heuristic algorithm should be made to construct better attacks.

Conclusions

In this paper, the *Strumok* cipher has been analyzed from the point of view of cipher initialization attacks and Guess-and-Determine attacks. A modified cipher initialization procedure is constructed for the *Strumok-512* cipher, which simplifies its analysis and, in particular, allows to prove that due to an error in the initialization procedure of the *Strumok-512* cipher, only 192 bits of the 256 bit vector initialization are efficiently used, leading to attacks with related keys. The security of the *Strumok* cipher against slide attacks with round shifts up to 7 has been proven. The use of slide attacks with rounds larger than 7 seems impractical.

Also, the Guess-and-Determine attack on the *Strumok-512* cipher is constructed with complexity 2^{448}, which violates the standard DSTU 8845 : 2019, which states that the complexity of the best attack on the *Strumok-512* cipher is equal to 2^{512}.

REFERENCES

[1] *Ukrainian National Standard DSTU 8845:2019. Cryptographic data security. Symmetric stream transformation algorithm* (2019). (In Ukranian).

[2] GORBENKO, I.—GORBENKO, Y.—TYMCHENKO, V.—KACHKO, O.: *Testing the speed of modern stream ciphers*, Computer science and cybersecurity (2018), 48–55.

[3] GORBENKO, I.—KUZNETSOV, O.—GORBENKO, Y.— ALEKSEYCHUK, A.—TYMCHENKO, V.: *Strumok keystream generator*. In: *2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, IEEE, May 2018. pp. 292–299.

[4] JIAO, L.—LI, Y.—HAO, Y.: *A guess-and-determine attack on SNOW-V stream cipher*, The Computer Journal **63** (2020), 1789–1812.

[5] KIRCANSKI, A.—YOUSSEF, A.: *On the sliding property of SNOW 3G and SNOW 2.0*, IET Information Security **5** (2011), no. 4, 199–206.

[6] KUZNETSOV, A. A.—POTII, O. V.—POLUYANENKO, N. A.— GORBENKO,Y. I.—KRYVINSKA, N.: *Stream Symmetric Cipher "Strumok"*. In: *Stream Ciphers in Modern Real-time IT Systems: Analysis, Design and Comparative Studies*, Cham: Springer International Publishing, November 2021, pp. 467–516.

[7] KUZNETSOV, O.—LUTSENKO, M.—IVANENKO, D.: *Strumok stream cipher: Specification and basic properties*. In: *2016 Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T)*, IEEE, October 2016, pp. 59–62.

Received September 30, 2022
Revised September 4, 2023
Accepted December 18, 2023
Publ. online June 10, 2024

Department of Mathematical Methods of Information Security
Institute of Physics and Technology
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
37, Prospect Beresteiskyi
03056–Kyiv
UKRAINE

E-mail: a.fesenko@kpi.ua