Algebraic Aspects of the Russian Hash Standard GOST R 34.11-2012

Oleksandr Kazymyrov, Valentyna Kazymyrova

Abstract

New GOST R 34.11-2012 standard has been recently selected by the Russian government to replace the old one. The algorithm is based on the hash function Stribog introduced in 2010. The high-level structure of the new hash function is similar to GOST R 34.11-94 with minor modifications. However, the compression function was changed significantly. Such a choice of the compression algorithm has been motivated by the Rjndael due to simplicity and understandable algebraic structure.

In this paper we consider a number of algebraic aspects of the GOST R 34.11. We show how one can express the cipher in AES-like form over the finite field $\mathbb{F}_{2^8}$, and consider some approaches that can be used for the fast software implementation.

Keywords: hash function, Stribog, GOST R 34.11-2012, field

1 Introduction

Until recently Russia has used a hash function defined by the standard GOST R 34.11-94 [1, 2]. Latest cryptanalytical results show that the standard has weaknesses from the theoretical point of view [3]. Therefore, the government forced to create a new cryptographically strong hash function.

In 2010 at RusCrypto'10 conference a prototype of a perspective hash function also known as "Stribog" [4, 5] was presented. The new algorithm is based on the modified Merkle-Damgård scheme with new compression function and digest sizes of 256 and 512 bits. In 2012 the hash function was accepted as the governmental standard GOST R 34.11-2012 [6, 7, 8]. It provides calculation procedure for any binary sequences used in cryptographic methods of information processing including techniques for providing data integrity and authenticity. This standard can be used for creation, operation and modernization of information systems for different purposes. At the same time the standard GOST R 34.10-2001 was replaced by the new one in 2012 taking into account the new hash algorithm.

The description method of hashing algorithm differs from the AES [10, 11]. It is oriented on engineers and programmers without strong mathematical background and is given in algorithm-like form [6]. Even the Stribog's specification does not give information about algebraic features and properties of basic operations. From the cryptanalytical point of view, it is necessary to have an algebraic structure for being able to find weaknesses and/or prove strengths of the algorithm.

In this paper we give a number of GOST R 34.11-2012 representations and consider an approach that could be applied to find the AES-like form over a finite field $\mathbb{F}_{2^8}$.

2 Description of the GOST R 34.11-2012

Hereinafter we assume that Stribog and GOST R 34.11-2012 are the same algorithms. GOST R 34.11-2012 specifies two iterative hash algorithms called Stribog-256 and Stribog-512 that process output message digest of 256 and 512 bits respectively. These algorithms differ in the initialization vector value and in the truncated message digest to 256 most significant bits (MSBs) in Stibog-256 case. Moreover the standard defines two more transformation that are addition modulo 2^{512} ($\boxplus$) and concatenation of two vectors A and B ($A||B$). The value of IV equals 0^{512} (all zero bits) and $(00000001)^{64}$ (64 bytes of 0x01 each) for Stribog-512 and Stribog-256 respectively.

It should be noted that the byte ordering is not specified in the standard. As in the previous standard bytes of information stored on a hard drive or transmitted to a channel have little-endian notation. That is, the message $M_2 = 0xFBE2E5 \dots E220E5D1$ from Appendix 2.2 [6] is stored on the disk in the from $M_2 = 0xD1E520E2 \dots E5E2FB$. Moreover, decoding the last string using the code page CP1251 (Windows-1251) gives "Се ветри, Стрибожи внуци, веютъ с моря стрелами на храбрыя плъкы Игоревы", which is a phrase from "The Tale of Igor's Campaign" [9]. Therefore, the description of the hash function is given in the form provided in the standard. In real applications endianness must be taking into account.

The hash algorithm consists of initialization, iterative and final stages. Figure 1 depicts general iterative structure of the hashing algorithm.

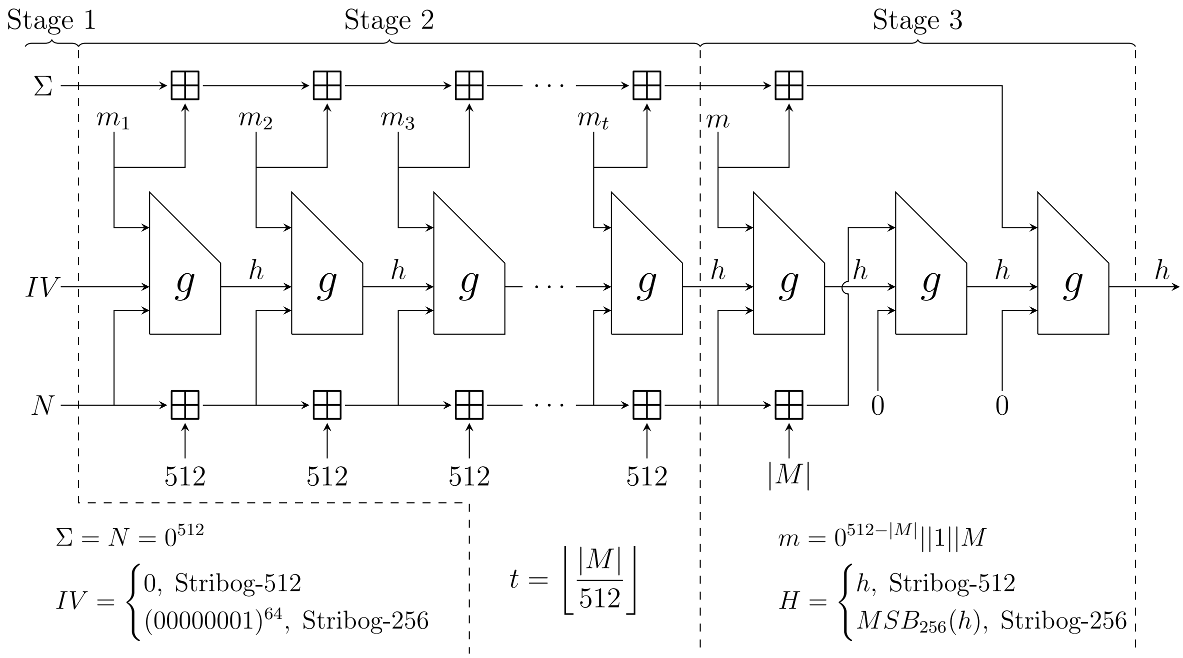

Figure 1: Stage Dividing of GOST R 34.11-2012

At the initialization stage (stage 1) the variables Σ, N and h assign the constant values $0, 0$ and IV respectively. At the next stage, the input message $M = M'||m_i$ divides into messages m_i, $1 \leq i \leq \left\lfloor \frac{|M|}{512} \right\rfloor$, of length 512 bits. Further, for each message m_i the iterative procedure based on a compression function $g_N(h, m)$ is applied. Finally, at the stage 3, consistent application of g_N with different parameters are made for the rest of the message M even if $|M| = 0$.

The standard GOST R 34.11-2012 specifies three main transformations S (SubBytes), P (Transposition) and L (MixColumns). These transformations (see Figure 2) underlie the following compression function $g_N : \mathbb{F}_2^{512} \times \mathbb{F}_2^{512} \mapsto$

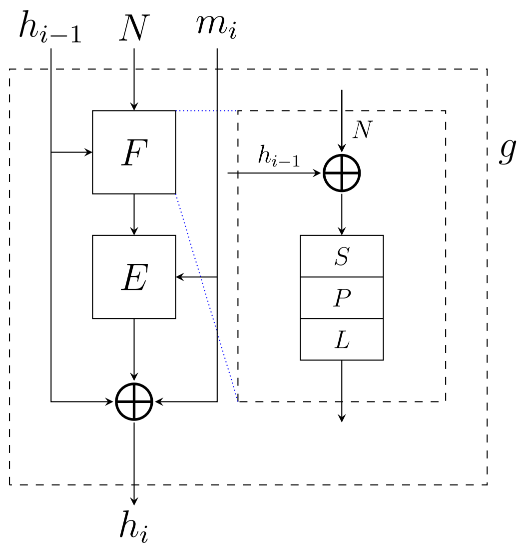

Figure 2: Compression Function of GOST R 34.11-2012

$\mathbb{F}_2^{512}$, $N \in \mathbb{F}_2^{512}$

$$g_N(h, m) = E(L \circ P \circ S(h \oplus N), m) \oplus h \oplus m, \quad h, m \in \mathbb{F}_2^{512}.$$

The E function is a block cipher of the form

$$E(K, m) = X[K_{13}] \circ \prod_{i=1}^{12} \left(L \circ P \circ S \circ X[K_i](m)\right).$$

The round keys K_i are calculated using the key schedule procedure with the following algorithm

$$K_i = L \circ P \circ S(K_{i-1} \oplus C_{i-1}),\ K_1 = K,\ i \in \{2, \ldots, 13\}.$$

In [5, 7] values of C_i are defined as the 512-bit constants (see Appendix A). The $X[K_i]$ operation is similar to AddRoundKey(K_i) of AES. The result of $X[K_i](A)$ is the bitwise XOR addition of round key K_i and input vector A.

As in AES the internal state of g_N can be represented as a byte matrix. However, in contrast to AES the Stribog's matrix is 8 by 8 bytes. The correspondence between the input vector B of 64 bytes and the state is presented in Figure 3.

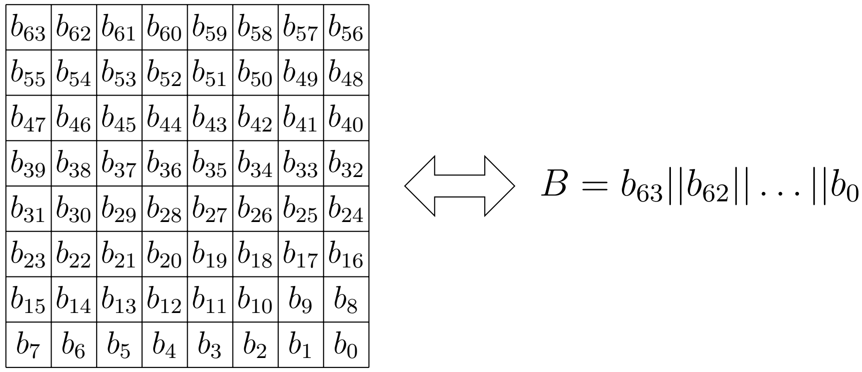

Figure 3: State Representation of Stribog

The S transformation is defined as the message partitioning into bytes followed by non-linear bijective mapping of each byte using substitution described in Appendix B. Clearly, the substitution of Stribog differs from the AES one. The maximum absolute value of the bias and the difference probability of the Stribog's S-box equal $\frac{7}{2^6}$ and $\frac{1}{2^5}$ respectively. Other properties are given in Appendix C.

The S transformation is the same as the SubBytes in AES and therefore has the same correspondence between input and output states (Figure 4).

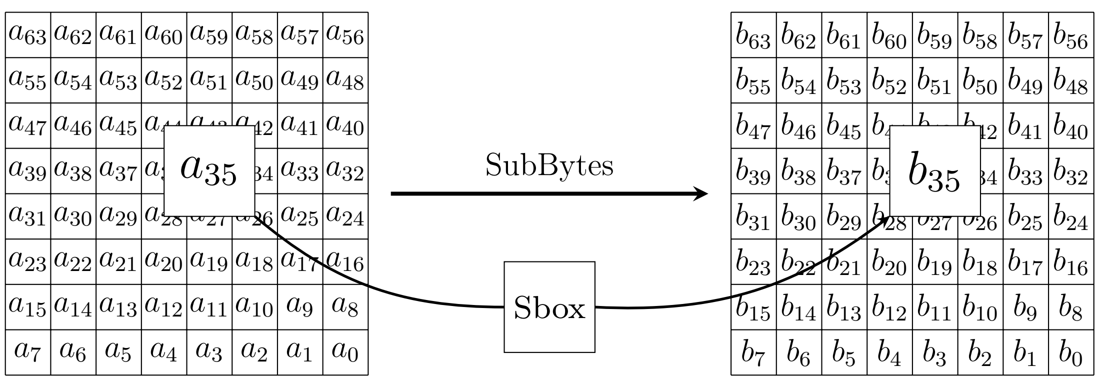

Figure 4: The S (SubBytes) Transformation

During the transformation P bits of the input message are grouped into bytes and are permuted in accordance with the permutation τ

$$\begin{aligned}\tau = \{&0, 8, 16, 24, 32, 40, 48, 56, 1, 9, 17, 25, 33, 41, 49, 57,\\ &2, 10, 18, 26, 34, 42, 50, 58, 3, 11, 19, 27, 35, 43, 51, 59,\\ &4, 12, 20, 28, 36, 44, 52, 60, 5, 13, 21, 29, 37, 45, 53, 61,\\ &6, 14, 22, 30, 38, 46, 54, 62, 7, 15, 23, 31, 39, 47, 55, 63\}.\end{aligned}$$

The similar transformation in the AES is ShiftRows. However, P transposes the matrix instead of shifting its rows (Figure 5).

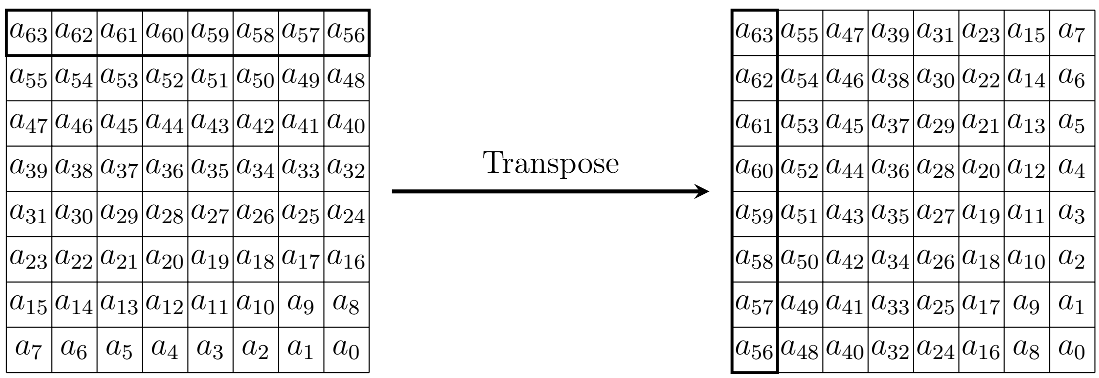

Figure 5: The P (Transposition) Transformation

The L transformation is based on a linear transformation l, which is given by the right multiplication by a fixed 64×64 matrix over the field $\mathbb{F}_2$

$$B = A \cdot M,$$

where A and B are input and output states respectively. Therefore, at the first step of L an input message is converted to the 64-bit vectors. Next, the transformation l applies for each vector (see Appendix D). At the last step, vector values obtained at the previous step are joint into an output message. Figure 6 depicts all these steps which are similar to the MixColumns transformation of AES.

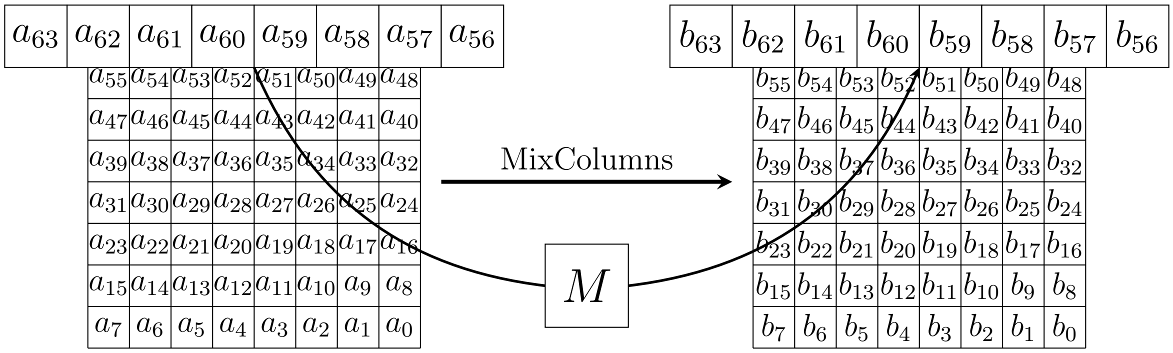

Figure 6: The L (MixColumns) Transformation

3 AES-like Representation of GOST R 34.11-2012

The description of hash functions, given in the previous section, significantly simplifies understanding of the principles underlying the algorithm compared to one given in the standard GOST R 34.11-2012. However, it does not allow to estimate the security aspects of the hashing algorithm. At the same time, the representation of g_N in AES-like form gives the opportunity to use mathematical tools that were created during last 15 years.

Since the state representations in the AES and Stribog are different, a reverse transformation must be applied at the first step to an input message. Suppose R is the transformation which return message with reversed bits. Obviously that $R^{-1} \circ R(x) = R \circ R(x) = x$. Then the compression function of GOST R 34.11-2012 can be performed by following three steps

- reverse input bits;
- AES-like transformations;
- reverse output bits.

The connection between input and output bytes is shown in Figure 7.

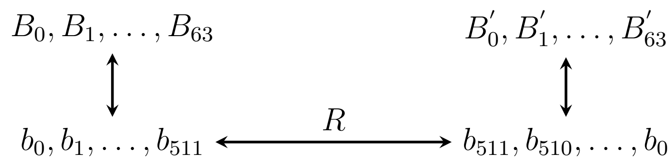

Figure 7: Reverse Transformation

The reverse transformation leads to changing of S, P, L and $X[K]$ transformations of the $g_N(h, m)$ function. Obviously, P and $X[K]$ do not need changes except applying R.

Since the S transformation is based on the constant substitution, applying the function $F'(x) = R \circ F \circ R(x)$, where F is the original S-box, to each byte gives a substitution for AES-like form (Appendix B). It is easy to see that vectorial Boolean functions F' and F are affine equivalent, therefore they have the same properties.

It is well-known that matrix multiplication over $\mathbb{F}_{2^8}$ has at least three forms

- representation over $\mathbb{F}_{2^n}$;
- representation over $\mathbb{F}_2$:
  - using matrix;
  - system of equations.

If a matrix is given over $\mathbb{F}_{2^n}$, then it is easy to find a representation over $\mathbb{F}_2$ for both system of equations and matrix forms. However, the reverse statement in general is not true because of a large amount of possible irreducible polynomials for large n. Nevertheless, for small fields all polynomials are known. There are only 30 irreducible polynomials for $n = 8$ [12].

Let $L : \mathbb{F}_{2^n} \mapsto \mathbb{F}_{2^n}$ be a linear function of the form [13]

$$L(x) = \sum_{i=0}^{n-1} \delta_i x^{2^i}.$$

For $\delta_i = 0$, $1 \le i < n$, L becomes

$$L(x) = \delta x.$$

This means that any multiplication mapping $\mathbb{F}_{2^n} \mapsto \mathbb{F}_{2^n}$ is a linear transformation of a vector space over $\mathbb{F}_2$ for specified basis. In [13] was shown that multiplication by arbitrary $\delta \in \mathbb{F}_{2^8}$ can be represented as multiplication on a matrix

$$\delta x = \begin{pmatrix} k_{0,0} & \cdots & k_{0,7} \\ k_{1,0} & \cdots & k_{1,7} \\ \vdots & \ddots & \vdots \\ k_{7,0} & \cdots & k_{7,7} \end{pmatrix} \cdot \begin{pmatrix} x_0 \\ x_1 \\ \ldots \\ x_7 \end{pmatrix}$$

where $x_i, k_{j,s} \in \mathbb{F}_2$. Using this representation any linear function $L : \mathbb{F}_{2^n} \mapsto \mathbb{F}_{2^m}$ can be converted to a matrix with the computation complexity $O(n)$. Further in [14] was proven that vice versa transformation can be done with the complexity of $O(n^3)$ field operations.

Thus, the algorithm of finding the matrix over F_{2^n} is as follows. For all possible irreducible polynomials convert all $n \times n$ bits submatrices to an element of the field and check MDS property of the resulting matrix.

The matrix over $\mathbb{F}_{2^8}$ with irreducible polynomial $f(x) = x^8+x^6+x^5+x^4+1$ received by the algorithm for Stribog is

$$M = \begin{pmatrix} 71 & 05 & 09 & B9 & 61 & A2 & 27 & 0E \\ 04 & 88 & 5B & B2 & E4 & 36 & 5F & 65 \\ 5F & CB & AD & 0F & BA & 2C & 04 & A5 \\ E5 & 01 & 54 & BA & 0F & 11 & 2A & 76 \\ D4 & 81 & 1C & FA & 39 & 5E & 15 & 24 \\ 05 & 71 & 5E & 66 & 17 & 1C & D0 & 02 \\ 2D & F1 & E7 & 28 & 55 & A0 & 4C & 9A \\ 0E & 02 & F6 & 8A & 15 & 9D & 39 & 71 \end{pmatrix}.$$

It should be noted that the binary matrix of Stribog additionally must be transposed [15].

Therefore, the L transformation becomes equivalent to MixColumns of AES and has the form

$$B = M \cdot A.$$

Suppose E^A, L^A, P^A, S^A are AES-like transformations for E, L, P, S respectively. Then it is easy to show (see Appendix E) that the modified $g_N(h, m)$ takes the form depicted in Figure 8.

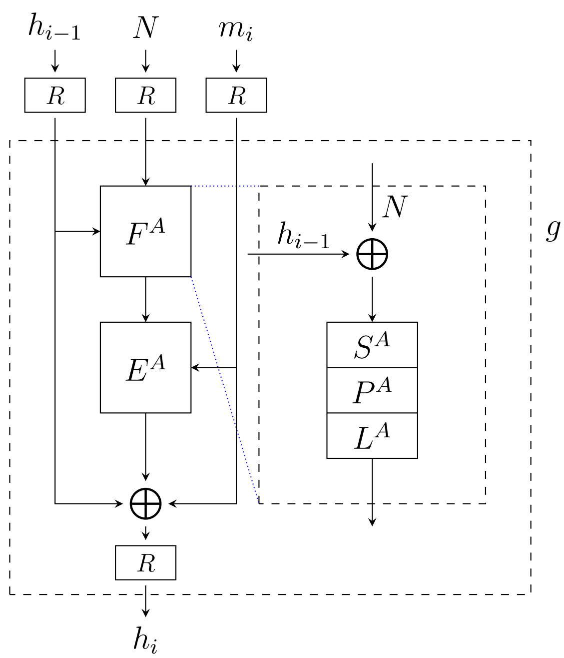

Figure 8: The Modified Compression Function for AES-like Representation

Since the calculation of block cipher E including key schedule procedure takes most of the time, fast implementation of this part of the hash function is needed for the maximum performance. The description in AES-like form gives access to use tables for increasing performance. Obviously, all optimization techniques described in [10] can be applied to the new standard. Various implementations of the hash function are given in [16].

4 Conclusions

Whole standard has been written in algorithm way and oriented on end developers. Shifting from functional and algorithmic description to logical and mathematical, which is more familiar for cryptographic primitives, allows us to estimate the security properties of Stribog. Our analysis shows that the algorithm of $g_N(h, m)$ is a modified version of AES with block and key lengths equal 512 bits. AES-like representation enables to prove resistant of the hash function to different types of attacks based on differential and linear cryptanalysis. Additionally, such a form shows that Stribog can be implemented by using tables.

References

[1] Information technology. Cryptographic Data Security. Hashing function. // GOST R 34.10-94, Gosudarstvennyi Standard of Russian Federation, Government Committee of the Russia for Standards. — Moscow, 1994. (In Russian)

[2] Dolmatov V. GOST R 34.11-94: Hash Function Algorithm. — http://tools.ietf.org/html/rfc5831, 10.03.2013.

[3] Florian Mendel et al. Cryptanalysis of the GOST Hash Function // Lecture Notes in Computer Science. — 2008. — V. 5157. — P. 162-178.

[4] Matyukhin D.V. et al. A perspective hashing algorithm. — Moscow: RusCrypto, 2010. (In Russian)

[5] Information technology. Cryptographic data security. Hash function. — http://infotecs.ru/laws/gost/proj/gost3411.pdf, 10.03.2013. (In Russian)

[6] Information technology. Cryptographic Data Security. Hash-functions. // GOST R 34.10-2012, Gosudarstvennyi Standard of Russian Federation, Government Committee of the Russia for Standards. — Moscow, 2012. (In Russian)

[7] Information technology. Cryptographic Data Security. Hash-functions. // GOST R 34.10-2012. — https://www.tc26.ru/en/GOSTR3411-2012/ENG_GOST_R_3411-2012_v1.pdf, 10.03.2013.

[8] Dolmatov V., Degtyarev A. GOST R 34.11-2012: Hash Function. — http://tools.ietf.org/html/rfc6986, 02.09.2013.

[9] The Tale of Igor's Campaign. — http://en.wikipedia.org/wiki/The_Tale_of_Igo
02.09.2013.

[10] Daemen J., Rijmen V. AES Proposal: Rijndael. — http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf, 10.03.2013.

[11] Announcing the ADVANCED ENCRYPTION STANDARD (AES) // Federal Information Processing Standards Publication 197, United States National Institute of Standards and Technology (NIST). — 2001.

[12] Lidl R., Niederreiter H. Finite Fields // Cambridge University Press. — 1997. — V. 20. Part 1. — 378 p.

[13] Budaghyan L., Kazymyrov O. Verification of Restricted EA-Equivalence for Vectorial Boolean Functions // Lecture Notes in Computer Science. — 2012. — V. 7369. — P. 108-118.

[14] Budaghyan L., Kazymyrov O. Verification of Restricted EA-Equivalence for Vectorial Boolean Functions // Information processing systems. — Kharkiv, 2013.

[15] Kazymyrov O. Representataions of MDS Matrices Over Finite Fields. — https://github.com/okazymyrov/MDS, 10.03.2013.

[16] Kazymyrov O. Implementations of the Stribog Hash Function. — https://github.com/okazymyrov/stribog, 10.03.2013.

A Constants Values for Key Schedule

The standard GOST R 34.11-2012 specifies the following 12 constants

$$\begin{aligned}
C_1 =& b1085bda1ecadae9ebcb2f81c0657c1f2f6a76432e45d016714eb88d7585c4fc\\
& 4b7ce09192676901a2422a08a460d31505767436cc744d23dd806559f2a64507;\\
C_2 =& 6fa3b58aa99d2f1a4fe39d460f70b5d7f3feea720a232b9861d55e0f16b50131\\
& 9ab5176b12d699585cb561c2db0aa7ca55dda21bd7cbcd56e679047021b19bb7;\\
C_3 =& f574dcac2bce2fc70a39fc286a3d843506f15e5f529c1f8bf2ea7514b1297b7b\\
& d3e20fe490359eb1c1c93a376062db09c2b6f443867adb31991e96f50aba0ab2;\\
C_4 =& ef1fdfb3e81566d2f948e1a05d71e4dd488e857e335c3c7d9d721cad685e353f\\
& a9d72c82ed03d675d8b71333935203be3453eaa193e837f1220cbebc84e3d12e;\\
C_5 =& 4bea6bacad4747999a3f410c6ca923637f151c1f1686104a359e35d7800fffbd\\
& bfcd1747253af5a3dfff00b723271a167a56a27ea9ea63f5601758fd7c6cfe57;\\
C_6 =& ae4faeae1d3ad3d96fa4c33b7a3039c02d66c4f95142a46c187f9ab49af08ec6\\
& cffaa6b71c9ab7b40af21f66c2bec6b6bf71c57236904f35fa68407a46647d6e;\\
C_7 =& f4c70e16eeaac5ec51ac86febf240954399ec6c7e6bf87c9d3473e33197a93c9\\
& 0992abc52d822c3706476983284a05043517454ca23c4af38886564d3a14d493;\\
C_8 =& 9b1f5b424d93c9a703e7aa020c6e41414eb7f8719c36de1e89b4443b4ddbc49a\\
& f4892bcb929b069069d18d2bd1a5c42f36acc2355951a8d9a47f0dd4bf02e71e;\\
C_9 =& 378f5a541631229b944c9ad8ec165fde3a7d3a1b258942243cd955b7e00d0984\\
& 800a440bdbb2ceb17b2b8a9aa6079c540e38dc92cb1f2a607261445183235adb;\\
C_{10} =& abbedea680056f52382ae548b2e4f3f38941e71cff8a78db1fffe18a1b336103\\
& 9fe76702af69334b7a1e6c303b7652f43698fad1153bb6c374b4c7fb98459ced;\\
C_{11} =& 7bcd9ed0efc889fb3002c6cd635afe94d8fa6bbbebab07612001802114846679\\
& 8a1d71efea48b9caefbacd1d7d476e98dea2594ac06fd85d6bcaa4cd81f32d1b;\\
C_{12} =& 378ee767f11631bad21380b00449b17acda43c32bcdf1d77f82012d430219f9b\\
& 5d80ef9d1891cc86e71da4aa88e12852faf417d5d9b21b9948bc924af11bd720.
\end{aligned}$$

The modified constants for AES-like representation are given below.

$$\begin{aligned}
C_1^A =& e0a2654f9aa601bbc4b22e336c2e6ea0a8cb0625105442458096e64989073ed2\\
& 3f23a1aeb11d728e680ba274c26e56f4f83ea60381f4d3d7975b53785bda108d;
\end{aligned}$$

$$
\begin{aligned}
C_2^A =&edd98d840e209e676ab3d3ebd845bbaa53e550db4386ad3a1a996b48d\\
&6e8ad598c80ad68f07aab8619d4c4504e577fcfebad0ef062b9c7f258f4b99551adc5f6;\\
C_3^A =&4d505d50af6978998cdb5e61c22f6d4390db4606ec5c93838d79ac092\\
&7f047cbdede948d28ae574fd1f8394afa7a8f60ac21bc56143f9c50e3f473d4353b2eaf;\\
C_4^A =&748bc7213d7d30448fec17c98557ca2c7dc04ac9ccc8ed1bae6bc0b74\\
&134eb95fcac7a16b5384eb9be3c3acc7ea17112bb278eba0587129f4b66a817cdfbf8f7;\\
C_5^A =&ea7f363ebf1ae806afc657957e456a5e6858e4c4ed00fffbc5af5ca4e\\
&2e8b3fdbdfff001ebac79ac52086168f838a8fec6c495363082fc5999e2e2b535d657d2;\\
C_6^A =&76be26625e02165facf2096c4ea38efd6d637d4366f84f502ded5938e\\
&d655ff363710f592d59fe183625428a9f2366b4039c0c5edcc325f69bcb5cb87575f275;\\
C_7^A =&c92b285cb26a6111cf523c4532a2e8ac20a05214c196e260ec3441b4a\\
&3d5499093c95e98cc7ce2cb93e1fd67e363799c2a9024fd7f61358a37a355776870e32f;\\
C_8^A =&78e740fd2bb0fe259b158a9aac43356cf423a58bd4b18b960960d949d\\
&3d4912f5923dbb2dc222d91787b6c398e1fed72828276304055e7c0e593c9b242daf8d9;\\
C_9^A =&db5ac4c18a22864e0654f8d3493b1c702a39e0655951d4de8d734ddbd\\
&02250012190b007edaa9b3c244291a4d85cbe5c7bfa68371b593229d9448c682a5af1ec;\\
C_{10}^A =&b739a219dfe32d2ec36ddca88b5f196c2f4a6edc0c36785ed2cc96f54\\
&0e6e7f9c086ccd85187fff8db1e51ff38e78291cfcf274d12a7541c4af6a001657b7dd5;\\
C_{11}^A =&d8b4cf81b32553d6ba1bf603529a457b1976e2beb8b35df7539d1257f\\
&78eb8519e6621288401800486e0d5d7ddd65f1b297f5ac6b363400cdf9113f70b79b3de;\\
C_{12}^A =&04ebd88f52493d1299d84d9babe82f5f4a1487115525b8e761338918b\\
&9f701bad9f9840c2b48041feeb8fb3d4c3c25b35e8d92200d01c84b5d8c688fe6e771ec.
\end{aligned}
$$

B Stribog's Lookup Tables

The following two tables describe the substitutions for the original GOST R 34.11-2012 and AES-like representations. All values in the table have hexadecimal notation.

Table 1: Substitution Box of GOST R 34.11-2012

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	FC	EE	DD	11	CF	6E	31	16	FB	C4	FA	DA	23	C5	04	4D
1	E9	77	F0	DB	93	2E	99	BA	17	36	F1	BB	14	CD	5F	C1
2	F9	18	65	5A	E2	5C	EF	21	81	1C	3C	42	8B	01	8E	4F
3	05	84	02	AE	E3	6A	8F	A0	06	0B	ED	98	7F	D4	D3	1F
4	EB	34	2C	51	EA	C8	48	AB	F2	2A	68	A2	FD	3A	CE	CC
5	B5	70	0E	56	08	0C	76	12	BF	72	13	47	9C	B7	5D	87
6	15	A1	96	29	10	7B	9A	C7	F3	91	78	6F	9D	9E	B2	B1
7	32	75	19	3D	FF	35	8A	7E	6D	54	C6	80	C3	BD	0D	57
8	DF	F5	24	A9	3E	A8	43	C9	D7	79	D6	F6	7C	22	B9	03
9	E0	0F	EC	DE	7A	94	B0	BC	DC	E8	28	50	4E	33	0A	4A
A	A7	97	60	73	1E	00	62	44	1A	B8	38	82	64	9F	26	41
B	AD	45	46	92	27	5E	55	2F	8C	A3	A5	7D	69	D5	95	3B
C	07	58	B3	40	86	AC	1D	F7	30	37	6B	E4	88	D9	E7	89
D	E1	1B	83	49	4C	3F	F8	FE	8D	53	AA	90	CA	D8	85	61
E	20	71	67	A4	2D	2B	09	5B	CB	9B	25	D0	BE	E5	6C	52
F	59	A6	74	D2	E6	F4	B4	C0	D1	66	AF	C2	39	4B	63	B6

Table 2: Substitution Box of GOST R 34.11-2012 for AES-like form

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	3F	FB	D7	E0	9F	E5	A8	04	97	07	AD	87	A0	B5	4C	9A
1	DF	EB	4F	0C	81	58	CF	D3	E8	3B	FD	B1	60	31	B6	8B
2	F3	7C	57	61	47	78	08	B4	C9	5E	10	32	C7	E4	FF	67
3	C4	3E	BF	11	D1	26	B9	7D	28	72	39	53	FE	96	C3	9C
4	BB	24	34	CD	A6	06	69	E6	0F	37	70	C1	40	62	98	2E
5	5F	6B	16	D6	3C	1C	1E	A4	8F	14	C8	55	B7	A5	63	F5
6	8C	C2	12	B8	F7	46	59	90	99	0D	6E	1F	F1	AA	51	2D
7	20	9D	73	E7	71	64	4D	36	FA	50	BA	A1	CB	A9	B0	C6
8	77	AF	2C	1A	18	E9	85	8E	EE	F0	0E	D8	21	A2	AE	65
9	23	9E	54	EC	38	1D	89	D9	6C	17	4E	CA	D0	C5	2A	66
A	76	15	13	35	3A	00	DE	D4	74	29	30	FC	56	7A	AC	2F
B	A3	44	5C	9B	80	F9	79	A7	B3	CC	ED	1B	2B	AB	BD	D2
C	88	95	8A	02	5A	CE	94	25	DB	7B	6A	92	75	49	BC	4B
D	5B	6F	45	27	42	41	F6	0B	DD	0A	E2	09	19	BE	01	43
E	68	93	D5	EF	84	22	E3	DA	5D	3D	48	7F	05	F4	7E	03
F	B2	C0	33	91	F2	82	8D	4A	83	52	E1	86	F8	DC	EA	6D

C S-box Properties of GOST R 34.11-2012

The comparison of Stribog and the AES substitutions is given in the following table. All properties presented in Table 3 were calculated according to the componet functions, which are the linear combinations (with non all-zero coeffcients) of the coordinate functions [13].

Table 3: Comparison of Stribog and AES Substitutions

Properties	Stribog	AES
Vectorial Boolean Function		
Balancedness	True	True
Nonlinearity	100	112
Absolute Indicator	96	32
Sum-of-squares Indicator	258688	133120
Propogation Criterion	0	0
Correlation Immunity	0	0
Minimum of Algebraic Degree	7	7
Resiliency	0	0
Strict Avalanche Criterion	False	False
Substitution		
Bijection	True	True
Maximum of Differential Table	8	4
Maximum of Approximation Table	28	16
Cycles Structure	252:243, 46:13	43:27, 242:87, 99:59, 124:81, 143:2
Algebraic Immunity	3(441)	2(39)

D The Constant Matrix for the l Transformation

The constant matrix is given in Table 4. Each value has hexadecimal notation and corresponds to a matrix row with index $i \cdot 4 + j$, $i = \{0, \dots, 15\}, j = \{0, \dots, 3\}$. For example, the row $21 = 5 \cdot 4 + 1$ is 8a174a9ec8121e5d.

E The Proof of AES-like Representation of $g_N(h, m)$

Taking into account all statements for L, P, S functions from Section 3, the modified E (E^A) takes the form

Table 4: The Constant Matrix of the Standard GOST R 34.11-2012

i \ j	0	1	2	3
0	8e20faa72ba0b470	47107ddd9b505a38	ad08b0e0c3282d1c	d8045870ef14980e
1	6c022c38f90a4c07	3601161cf205268d	1b8e0b0e798c13c8	83478b07b2468764
2	a011d380818e8f40	5086e740ce47c920	2843fd2067adea10	14aff010bdd87508
3	0ad97808d06cb404	05e23c0468365a02	8c711e02341b2d01	46b60f011a83988e
4	90dab52a387ae76f	486dd4151c3dfdb9	24b86a840e90f0d2	125c354207487869
5	092e94218d243cba	8a174a9ec8121e5d	4585254f64090fa0	accc9ca9328a8950
6	9d4df05d5f661451	c0a878a0a1330aa6	60543c50de970553	302a1e286fc58ca7
7	18150f14b9ec46dd	0c84890ad27623e0	0642ca05693b9f70	0321658cba93c138
8	86275df09ce8aaa8	439da0784e745554	afc0503c273aa42a	d960281e9d1d5215
9	e230140fc0802984	71180a8960409a42	b60c05ca30204d21	5b068c651810a89e
A	456c34887a3805b9	ac361a443d1c8cd2	561b0d22900e4669	2b838811480723ba
B	9bcf4486248d9f5d	c3e9224312c8c1a0	effa11af0964ee50	f97d86d98a327728
C	e4fa2054a80b329c	727d102a548b194e	39b008152acb8227	9258048415eb419d
D	492c024284fbaec0	aa16012142f35760	550b8e9e21f7a530	a48b474f9ef5dc18
E	70a6a56e2440598e	3853dc371220a247	1ca76e95091051ad	0edd37c48a08a6d8
F	07e095624504536c	8d70c431ac02a736	c83862965601dd1b	641c314b2b8ee083

$$E^A(K,m) = \big(R\circ X[K_{13}^A]\circ R\big)\circ \prod_{i=2}^{12}\Big(\big(R\circ L^A\circ R\big)\circ\big(R\circ P^A\circ R\big)\circ$$
$$\circ\big(R\circ S^A\circ R\big)\circ\big(R\circ X[K_i^A]\circ R\big)\Big)\circ\Big(\big(R\circ L^A\circ R\big)\circ$$
$$\circ\big(R\circ P^A\circ R\big)\circ\big(R\circ S^A\circ R\big)\circ\big(R\circ X[K_1^A]\circ R(m)\big)\Big) = R\circ X[K_{13}^A]\circ$$
$$\circ\prod_{i=2}^{12}\big(L^A\circ P^A\circ S^A\circ (X[K_i^A]\big)\circ\big(L^A\circ P^A\circ S^A\circ X[K_1^A]\circ R(m)\big)\,.$$

In fact, the message m is reversed at previous steps before calling the function $g_N(h,m)$. The final R is applied for the result of $g_N(h,m)$. Thus, the final algorithm of E^A has the form

$$E^A(K,m) = X[K_{13}^A]\circ\prod_{i=1}^{12}\big(L^A\circ P^A\circ S^A\circ X[K_i^A](m)\big)\,.$$

The round keys K_i^A are calculated using constants C_i^A (see. Appendix A)

$$K_i^A = L^A \circ P^A \circ S^A(K_{i-1}^A \oplus C_{i-1}^A),\ K_1^A = K^A,\ i \in \{2, \dots, 13\}.$$

All of the above lead to the modification of the whole function $g_N(h, m)$ (Figure 8)

$$K^A = L^A \circ P^A \circ S^A(R(h) \oplus R(N))$$
$$g_N(h, m) = R \circ \left(E(K^A, R(m)) \oplus R(h) \oplus R(m)\right).$$