GIỚI THIỆU VỀ THUẬT TOÁN MÃ HÓA KUZNYECHIK CỦA LIÊN BANG NGA

Bảo mật và an toàn thông tin, trong đó kỹ thuật mật mã đóng vai trò then chốt, là yếu tố tiên quyết để triển khai các hoạt động giao dịch điện tử. Do đó, việc chuẩn hóa các thuật toán mật mã sử dụng cho lĩnh vực kinh tế - xã hội luôn được các nước quan tâm, cập nhật và bổ sung. Các mã khối Magma và Kuznyechik được công bố trong tiêu chuẩn GOST R 34.12-2015 của Liên bang Nga. Bài báo này tổng hợp ngắn gọn về nguyên lý thiết kế và độ an toàn kháng lại các tấn công thám mã của thuật toán mã hóa Kuznyechik.

Trần Hồng Thái, Nguyễn Văn Long, Hoàng Đình Linh
Viện Khoa học - Công nghệ mật mã

1. GIỚI THIỆU CHUNG

Mã khối Kuznyechik là một thuật toán xử lý với kích thước khối dữ liệu 128 bit và kích thước khóa 256. Kuznyechik dựa trên cấu trúc mạng thay thế - hoán vị (SPN), đặc biệt là lược đồ khóa sử dụng mạng Feistel. Các thành phần chính của mã khối này gồm S-hộp với kích cỡ đầu vào 8 bit và đầu ra 8 bit, các phép biến đổi tuyến tính có tính khuếch tán tốt và các biến đổi đơn giản khác như phép cộng XOR. Chuẩn GOST R 34.12-2015 đã được chính thức phê duyệt ngày 19/6/2015 và có hiệu lực từ ngày 01/01/2016.

2. MÔ TẢ KUZNYECHIK

Kuznyechik là một mã khối có cấu trúc SPN biến đổi khối bản rõ 128-bit (ký hiệu P) thành khối bản mã 128-bit (ký hiệu C). Mã pháp sử dụng một khoá 256-bit (ký hiệu là K) để tạo ra 11 khoá vòng (ký hiệu lần lượt là $K_0, K_1, \ldots, K_{10}$), mỗi khoá vòng có kích cỡ 128-bit.

2.1 CÁC KÝ HIỆU

V^* :Tập hợp tất cả các chuỗi nhị phân độ dài hữu hạn, gồm cả chuỗi trống

V_s :Tập tất cả các chuỗi nhị phân độ dài s, trong đó s là số nguyên không âm

$|A|$:Độ dài của chuỗi $A \in V^*$ (nếu A – chuỗi trống, thì $|A| = 0$);

$A||B$:Nối của hai chuỗi $A, B \in V^*$, tức là một chuỗi từ $V_{|A|+|B|}$, trong đó chuỗi con cùng các thành phần có chỉ số lớn từ $V_{|A|}$ trùng với chuỗi A và chuỗi con cùng với các thành phần có chỉ số nhỏ từ $V_{|B|}$ trùng với với chuỗi B;

$\oplus$: Phép cộng modulo 2 theo từng thành phần của hai chuỗi nhị phân có độ dài như nhau;

Z_{2^s}: Vành thặng dư theo modulo 2^s;

$GF(2^8)$: Trường hữu hạn $GF(2)[x]/p(x)$, trong đó $p(x) = x^8 + x^7 + x^6 + x + 1 \in GF(2)[x]$;

Vec_s: $Z_{2^s} \rightarrow V_s$: Ánh xạ song ánh chuyển tương ứng một phần tử của vành Z_{2^s} sang biểu diễn nhị phân của nó, tức là chuyển phần tử bất kỳ $z \in Z_{2^s}$, mà được biểu diễn ở dạng $z = z_0 + 2 \cdot z_1 + \ldots + 2^{s-1} \cdot z_{s-1}$, trong đó $z_i \in \{0, 1\}$, $i = 0, 1, .., s - 1$ thành dạng chuỗi nhị phân dạng $z_{s-1}||\ldots.||z_1||z_0$;

Int_s: $V_s \rightarrow Z_{2^s}$: Ánh xạ ngược với ánh xạ Vec_s, tức là $\text{Int}_s = \text{Vec}_s^{-1}$;

$\Delta: V_8 \rightarrow GF(2^8)$: Ánh xạ song ánh, chuyển tương ứng một chuỗi nhị phân từ V_8 sang một phần tử của trường $GF(2^8)$

$\nabla: GF(2^8) \rightarrow V_8$: Ánh xạ ngược với ánh xạ Δ, tức là chuyển một phần tử của trường $GF(2^8)$ sang chuỗi nhị phân V_8

$\Phi\ \Psi$: Tổ hợp của hai ánh xạ, trong đó ánh xạ Ψ được thực hiện trước;

2.2. CÁC THAM SỐ

• S-hộp π là phép biến đổi phi tuyến thực hiện thay thế một khối 8 bit bởi một khối 8 bit khác được mô tả ở dạng bảng tra 256 phần tử như sau:

$\pi[256] = \{252, 238, 221, 17, 207, 110, 49, 22, 251,$

196, 250, 218, 35, 197, 4, 77, 233, 119, 240, 219, 147, 46, 153, 186, 23, 54, 241. 187, 20, 205, 95, 193, 249, 24, 101, 90, 226, 92, 239, 33, 129, 28, 60, 66, 139, 1, 142, 79, 5, 132, 2, 174, 227, 106, 143, 160, 6, 11, 237, 152, 127, 212, 211, 31, 235, 52, 44, 81, 234, 200, 72, 171, 242, 42, 104, 162, 253, 58, 206, 204, 181, 112, 14, 86, 8, 12, 118, 18, 191, 114, 19, 71, 156, 183, 93, 135, 21, 161, 150, 41, 16, 123, 154, 199, 243, 145, 120, 111, 157, 158, 178, 177, 50, 117, 25, 61, 255, 53, 138, 126, 109, 84, 198, 128, 195, 189, 13, 87, 223, 245, 36, 169, 62, 168, 67, 201, 215, 121, 214, 246, 124, 34, 185, 3, 224, 15, 236, 222, 122, 148, 176, 188, 220, 232, 40, 80, 78, 51, 10, 74, 167, 151, 96, 115, 30, 0, 98, 68, 26, 184, 56, 130, 100, 159, 38, 65, 173, 69, 70, 146, 39, 94, 85, 47, 140, 163, 165, 125, 105, 213, 149, 59, 7, 88, 179, 64, 134, 172, 29, 247, 48, 55, 107, 228, 136, 217, 231, 137, 225, 27, 131, 73, 76, 63, 248, 254, 141, 83, 170, 144, 202, 216, 133, 97, 32, 113, 103, 164, 45, 43, 9, 91, 203, 155, 37, 208, 190, 229, 108, 82, 89, 166, 116, 210, 230, 244, 180, 192, 209, 102, 175, 194, 57, 75, 99, 182}.

• Biến đổi tuyến tính được cho bởi ánh xạ ℓ: $V_8^{16} \to V_8$, được xác định như sau:

$\ell(a15, \ldots, a0) = \nabla\{148 \cdot \Delta(a_{15}) + 32 \cdot \Delta(a_{14}) + 133 \cdot \Delta(a_{13}) + 16 \cdot \Delta(a_{12}) + 194 \cdot \Delta(a_{11}) + 192 \cdot \Delta(a_{10}) + 1 \cdot \Delta(a_9) + 251 \cdot \Delta(a_8) + 1 \cdot \Delta(a_7) + 192 \cdot \Delta(a_6) + 194 \cdot \Delta(a_5) + 16 \cdot \Delta(a_4) + 133 \cdot \Delta(a_3) + 32 \cdot \Delta(a_2) + 148 \cdot \Delta(a_1) + 1 \cdot \Delta(a_0)\}$

với mọi $a_i \in V_8$, $i = 0, 1, \ldots, 15$, trong đó các phép toán cộng và nhân được thực hiện trong trường $GF(2^8)$, các hằng số là các giá trị của trường như đã định nghĩa ở trên.

2.3. CÁC PHÉP BIẾN ĐỔI

Các biến đổi sau được sử dụng cho thuật toán mã hoá và thuật toán giải mã:

X[k]: $V_{128} \to V_{128}$: X[k](α) = k $\oplus$ a, trong đó k, $a \in V_{128}$;

S: $V_{128} \to V_{128}$:$S(a) = S(a_{15}||\ldots ||a_0) = \pi(a_{15})||\ldots ||\pi(a_0)$, trong đó $a = a_{15}||\ldots||a_0 \in V_{128}$, $a_i \in V_8$, $i = 0, 1, .., 15$;

S^{-1} : $V_{128} \to V_{128}$: Biến đổi ngược của biến đổi S, ví dụ: nó có thể được tính bằng cách sau:

$S^{-1}(a) = S^{-1}(a_{15}||\ldots ||a_0) = \pi^{-1}(a_{15})|| \ldots ||\pi^{-1}(a_0)$, trong đó $a = a_{15}||\ldots.||a_0 \in V_{128}$, $a_i \in V_8$, $i = 0, 1, \ldots, 15$, π^{-1} là phép thế ngược (nghịch đảo) với phép thế π;

R: $V_{128} \to V_{128}$: $R(a) = R(a_{15}||\ldots||a_0) = \ell(a_{15},\ldots, a_0)||a_{15}||\ldots||a_1$, trong đó $a = a_{15}||\ldots.||a_0 \in V_{128}$, $a_i \in V_8$, $i = 0, 1, \ldots, 15$;

L: $V_{128} \to V_{128}$: $L(a) = R_{16}(a)$, trong đó $a \in V_{128}$;

R^{-1}: $V_{128} \to V_{128}$: Biến đổi ngược với biến đổi R, ví dụ: nó có thể được tính bằng cách sau:

$R^{-1}(a) = R^{-1}(a_{15}||\ldots||a_0) = a_{14}||a_{13}||\ldots ||a_0||\ell(a_{14},a_{13}, \ldots, a_0, a_{15})$, trong đó $a = a_{15}||\ldots||a_0 \in V_{128}$, $a_i \in V_8$, $i = 0, 1, \ldots, 15$;

L^{-1}: $V_{128} \to V_{128}$: $L^{-1}(a) = (R^{-1})^{16}(a)$,

F[k]: V_{128} x $V_{128} \to V_{128}$ x V_{128}: $F[k](a_1, a_0) = LSX[k](a_1) \oplus a_0, a_1$, trong đó k, $a_0, a_1 \in V_{128}$

2.4. THUẬT TOÁN LƯỢC ĐỒ KHÓA

Thuật toán lược đồ khóa sử dụng các hằng số vòng $C_i \in V_{128}$, $i = 1, 2, .., 32$, chúng được xác định như sau:

$$C_i = L(\text{Vec}_{128}(i)),\ i = 1, 2, .., 32.$$

Các khóa vòng $K_i \in V_{128}$, $i = 1, 2, .., 10$, được tạo ra trên cơ sở khóa $K = k_{255}||\ldots||k_0 \in V_{256}$, $k_i \in V_1$, $i = 0, 1, .., 255$, và được xác định bằng các đẳng thức:

$$K_1 = k_{255}||\ldots||k_{128};$$
$$K_2 = k_{127}||\ldots||k_0;$$

$(K_{2i+1}, K_{2i+2}) = F[C_{8(i-1)+8}]\ldots F[C_{8(i-1)+1}](F_{2i-1}, K_{2i})$, $i = 1, 2, 3, 4$.

2.5. THUẬT TOÁN MÃ HÓA

Thuật toán mã hoá phụ thuộc vào giá trị của các khóa vòng $K_i \in V_{128}$, $i = 1, 2, .., 10$, thực hiện lặp 9 lần hàm vòng (gồm các biến đổi cơ sở L, S, X trên tập V_{128}) như sau: $E_{K_1 K_2 \ldots K_{10}}(a) = X[K_{10}]LSX[K_9]\ldots. LSX[K_2]LSX[K_1](a)$, với $a \in V_{128}$ là khối bản rõ đầu vào.

2.6. THUẬT TOÁN GIẢI MÃ

Thuật toán giải mã phụ thuộc vào giá trị của các khóa vòng $K_i \in V_{128}$, $i = 1, 2, .., 10$, thực hiện phép thế $D_{K_1 K_2 \ldots K_{10}}$, nó được xác định trên tập V_{128} như sau:

$(a) = X[K_1]S^{-1}L^{-1}X[K_2]\ldots. S^{-1}L^{-1}X[K_9]S^{-1}L^{-1}X[K_{10}](a)$, với $a \in V_{128}$ là khối bản mã đầu vào.

3. MỘT SỐ ĐÁNH GIÁ ĐỘ AN TOÀN CỦA KUZNYECHIK

Độ an toàn của thuật toán Kuznyechik được đánh giá qua việc phân tích cấu trúc thiết kế kháng lại các tấn công mạnh hiện nay (thám mã vi sai, thám mã tuyến tính) và các kết quả tấn công mới nhất lên thuật toán này.

3.1. KHẢ NĂNG KHÁNG THÁM MÃ VI SAI VÀ THÁM MÃ TUYẾN TÍNH CỦA THUẬT TOÁN KUZNYECHIK

Để đánh giá khả năng kháng lại thám mã vi sai và thám mã tuyến tính của một mã pháp ta

cần chỉ ra rằng xác suất vi sai cực đại hoặc xác suất tuyến tính cực đại của mã pháp là đủ nhỏ. Vì độ phức tạp tính toán để thực hiện tấn công tỷ lệ nghịch với đại lượng trên, nên điều đó đồng nghĩa với độ phức tính toán là đủ lớn và đủ an toàn trước các tấn công nêu trên. Ta cần phải chỉ ra rằng độ phức tạp tính toán để thực hiện tấn công lớn hơn tấn công vét cạn khóa (trong trường hợp này là 2^{256}).

Thuật toán Kuznyechik được thiết kế dựa trên mạng SPN, trong đó tầng tuyến tính là 16 vòng hoạt động của thanh ghi dịch tuyến tính có phản hồi dạng Fibonacci. Tầng tuyến tính này có thể biểu diễn tương đương với một phép nhân bên trái của véc tơ hàng với ma trận tuyến tính kích thước 16×16. Trong [1] các tác giả đã chứng tỏ ma trận tuyến tính được sử dụng trong thuật toán Kuznyechik này là một ma trận MDS trên trường $GF(2^8)$.

Với việc sử dụng tầng tuyến tính là một ma trận MDS có kích thước 16×16 trên trường $GF(2^8)$, có mệnh đề sau:

Mệnh đề 1: *Hai vòng liên tiếp của mã khối Kuznyechik có ít nhất là 17 hộp thế chủ động vi sai (tuyến tính).*

Với mục đích giới thiệu thuật toán, nên chúng tôi bỏ qua chứng minh của mệnh đề trên. Theo đó, ta có thể tính được số S-hộp chủ động qua các vòng mã hóa. Trong đó, S-hộp chủ động vi sai (hoặc tuyến tính) là S-hộp có sai khác (hoặc mặt che tuyến tính) đầu vào tương ứng với S-hộp đó khác không.

Bảng 1 biểu diễn sự lan truyền các S-hộp chủ động qua các vòng (tính cả S-hộp chủ động ở đầu vào, đầu ra và ở các vòng trung gian). Số các S-hộp chủ động sẽ tăng lên khi số lượng vòng mã tăng.

Bảng 1. Số S-hộp chủ động qua các vòng mã hóa của Kuznyechik và AES

Vòng	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Kuznyechik	1	17	18	34	35	51	52	68	69	85				
AES	1	5	9	25	26	30	34	50	51	55	59	75	76	80

Giả sử ở vòng 1 của Kuznyechik có một S-hộp chủ động, qua vòng 2 sẽ có tổng cộng 17 S-hộp chủ động, và đến vòng 10 sẽ là 85 S-hộp chủ động. Điều này có được là nhờ hiệu ứng khuếch tán của phép biến đổi tuyến tính của mã pháp. Theo đó, với 10 vòng mã hóa của Kuznyechik đã đạt được độ an toàn tương ứng với 14 vòng mã của thuật toán AES. S-hộp của Kuznyechik được phân tích trong [2], trong đó xác suất vi sai cực đạ DP_{max} và tuyến tính cực đại LP_{max} của S-hộp này tương ứng là:

$$DP_{max}=(8/256)=2^{-5}$$

và

$$LP_{max}=(28/128)^2=2^{-4.81}$$

Kết hợp với Bảng 1, ta có đánh giá về khả năng kháng lại thám mã vi sai và tuyến tính (*được ước lượng theo nghĩa về độ phức tạp tính toán, đại lượng này tỷ lệ nghịch với giá trị xác suất vi sai/tuyến tính cực đại*) của mã pháp Kuznyechik và so sánh với thuật toán AES như Bảng 2. Độ phức tạp tính toán để có thể tấn công vi sai/tuyến tính với n vòng của mã pháp ước lượng là lớn hơn giá trị $1/(DP_{max})^{\text{số S-hộp chủ động}}$ hoặc $1/(LP_{max})^{\text{số S-hộp chủ động}}$ tương ứng.

Từ đây có thể thấy rằng thuật toán Kuznyechik với số vòng 10 là an toàn trước thám mã vi sai và thám mã tuyến tính với mức khóa có độ dài 256 bit.

Bảng 2. Độ an toàn tổng thể kháng lại thám mã vi sai và tuyến tính của Kuznyechik và AES

Thuật toán	Độ dài khóa	Thám mã vi sai	Thám mã tuyến tính
Kuznyechik	256	$2^{460.275}$ (6 vòng $2^{276.165}$)	$2^{408.85}$ (8 vòng $2^{327.08}$)
AES-256	256	2^{480} (8 vòng 2^{300})	2^{480} (8 vòng 2^{300})

3.2. CÁC TẤN CÔNG THÁM MÃ KHÁC LÊN KUZNYECHIK

Dưới đây là một số kết quả phân tích và tấn công thám mã lên Kuznyechik. Tháng 4/2015, Riham AlTawy và Amr M. Youssef [3] đã mô tả một tấn công gặp ở giữa (meet-in-the-middle attack) lên 5 vòng Kuznyechik. Tấn công này cho phép khôi phục khoá với độ phức tạp thời gian 2^{140}, độ phức tạp bộ nhớ 2^{153}, và độ phức tạp dữ liệu 2^{113}.

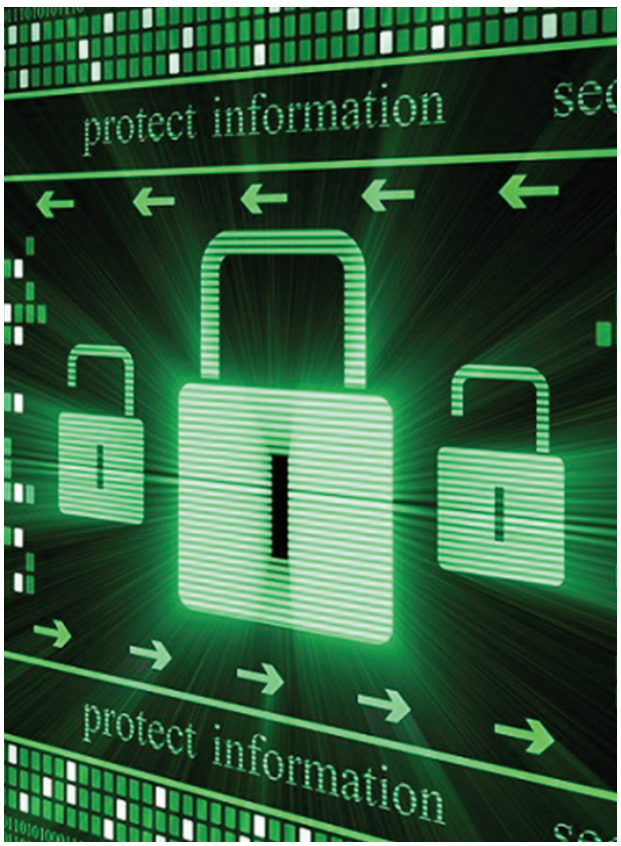
Sau đó, Riham AlTawy, Onur Duman, và Amr M. Youssef [4] đã công bố hai tấn công gây lỗi lên Kuznyechik và từ đó chỉ ra sự quan trọng trong việc bảo vệ cài đặt của mã pháp. Cụ thể, các tác giả đã trình bày 2 tấn công phân tích lỗi lên Kuznyechik trong hai cài đặt khác nhau. Tấn công đầu tiên là một tấn công gây lỗi vi sai mà sử dụng mô hình lỗi byte ngẫu nhiên. Trong tấn công này, các tác giả khai thác một biểu diễn tương đương của hàm vòng cuối cùng mà khi đó chúng ta bỏ qua tác động của sự khuếch tán tối ưu của biến đổi tuyến tính cuối cùng. Sự điều chỉnh này cho phép một tấn công khôi phục khoá thực hành được và hiệu quả chỉ với 4 lỗi. Tấn công thứ 2 là một phân tích lỗi không có hiệu quả 4 giai đoạn của Kuznyechik khi khai thác các S-hộp bí mật. Đầu tiên, các tác giả khôi phục một số tập các ứng viên cho các S-hộp bí mật và khoá chính tương ứng của chúng. Sau đó, các tác giả lọc các ứng viên này để

thu được khoá chính và các S-hộp đúng bằng cách kiểm tra đối với một cặp bản rõ mã đã biết. Các tấn công này hoạt động khi giả sử rằng các S-hộp bí mật được sử dụng lại trong lược đồ khoá.

Năm 2016, Alex Biryukov, Leo Perrin, và Aleksei Udovenko [5] đã công bố một bài báo trong đó chỉ ra rằng, các S-hộp của Kuznyechik và Streebog không được tạo một cách giả ngẫu nhiên và bằng cách sử dụng một thuật toán ẩn thì họ có thể khôi phục ngược. Tuy nhiên, các kết quả trên chưa ảnh hưởng đến độ an toàn của của Kuznyechik, chúng ý nghĩa lớn trong việc tăng tốc độ trong thực thi phần cứng của thuật toán này.

Năm 2017, Biryukov [6] đã đề xuất một phương pháp thám mã mới với tên gọi là thám mã đại số-đa tập lên Kuznyechik sử dụng các tính chất chia-tích phân bậc đại số của các thành phần. Theo đó, các tác giả đã chỉ ra rằng tấn công này có thể áp dụng được lên 7 vòng của mã pháp Kuznyechik.

Gần đây, nhóm tác giả Youssef [7] công bố một tấn công gặp nhau ở giữa cải tiến mới lên 6 vòng của Kuznyechik. Tấn công này khai thác tính chất có cấu trúc của ma trận MDS trong mã pháp. Theo đó, các tác giả tìm được các quan hệ giữa các tập con đầu vào và đầu ra của biến đổi tuyến tính này, sử dụng các quan hệ này để tìm các bộ phân biệt tốt và một số kỹ thuật tối ưu khác nhằm làm giảm độ phức tạp của tấn công.



4. KẾT LUẬN

Cho đến nay đã có rất nhiều công trình nghiên cứu, phân tích và đánh giá về mã khối Kuznyechik. Một số tấn công điển hình gồm kết quả của nhóm R. AlTawy và A. M. Youssef và nhóm A. Biryukov cùng cộng sự. Nhóm tác giả R. AlTawy đã chỉ ra một tấn công gặp ở giữa (meet-in-the-middle attack) lên 5 vòng Kuznyechik, cho phép khôi phục khoá với độ phức tạp thời gian 2^{140}, độ phức tạp bộ nhớ 2^{153}, và độ phức tạp dữ liệu 2^{113}. Nhóm tác giả A. Biryukov chỉ ra rằng các S-hộp của Kuznyechik và Streebog không được tạo một cách giả ngẫu nhiên và bằng cách sử dụng một thuật toán ẩn thì họ có thể khôi phục ngược. Tuy nhiên, có thể thấy rõ ràng rằng hầu hết các tấn công này mới chỉ dừng ở mức phân tích với số vòng rút gọn mà không ảnh hưởng tới độ an toàn của thuật toán Kuznyechik đầy đủ.

TÀI LIỆU THAM KHẢO

[1].Borisenko, N., V. Nguyen, and A. Bulygin. Developing Algorithm for Software and Hardware Implementation of Large Size Linear Mapping. in 2nd Workshop on Current Trends in Cryptology (CTCrypt 2013).–June. 2013.

[2].Kazymyrov, O. and V. Kazymyrova, Algebraic Aspects of the Russian Hash Standard GOST R 34.11-2012. IACR Cryptology ePrint Archive, 2013. 2013: p. 556.

[3].AlTawy, R. and A.M. Youssef, A meet in the middle attack on reduced round Kuznyechik. IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences, 2015. 98(10): p. 2194-2198.

[4].AlTawy, R., O. Duman, and A.M. Youssef, Fault analysis of Kuznyechik. Математические вопросы криптографии, 2016. 7(2): p. 21-34.

[5].Biryukov, A., L. Perrin, and A. Udovenko. Reverse-engineering the S-box of Streebog, Kuznyechik and STRIBOBr1. in Annual International Conference on the Theory and Applications of Cryptographic Techniques. 2016. Springer.

[6].Biryukov, A., D. Khovratovich, and L. Perrin, Multiset-algebraic cryptanalysis of reduced Kuznyechik, Khazad, and secret SPNs. IACR Transactions on Symmetric Cryptology, 2017. 2016(2): p. 226-247.

[7].Tolba, M. and A.M. Youssef. Improved Meet-in-the-Middle Attacks on Reduced Round Kuznyechik. in International Conference on Information Security and Cryptology. 2017. Springer.