УДК 621.391:519.2:519.7

Л.В. КОВАЛЬЧУК
Фізико-технічний інститут Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського»; Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України, Київ, Україна,
e-mail: *lusi.kovalchuk@gmail.com.*

І.В. КОРЯКОВ
Товариство з обмеженою відповідальністю «Науково-впроваджувальна фірма Криптон», Київ, Україна,
e-mail: *ikor@i.ua.*

А.М. ОЛЕКСІЙЧУК
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Україна,
e-mail: *alex-dtn@ukr.net.*

KRIP: ВИСОКОШВИДКІСНИЙ АПАРАТНО-ОРІЄНТОВАНИЙ ПОТОКОВИЙ ШИФР, ПОБУДОВАНИЙ НА ОСНОВІ НЕАВТОНОМНОГО НЕЛІНІЙНОГО РЕГІСТРУ ЗСУВУ

Анотація. Запропоновано алгоритм потокового шифрування, побудований на основі неавтономного нелінійного регістру зсуву довжини 2 над алфавітом потужності 2^{256}. Цей регістр функціонує аналогічно шифру Фейстеля з раундовою функцією, що використовується в алгоритмі шифрування Kalyna. Показано, що за стійкості на рівні 2^{256} шифр Krip забезпечує чотирикратний виграш у швидкодії порівняно з прийнятим стандартом потокового шифрування України та майже двадцятикратний порівняно з сучасним алгоритмом шифрування Espresso.

Ключові слова: потоковий алгоритм шифрування, схема Фейстеля, нелінійний регістр зсуву, генератор псевдовипадкових послідовностей, алгебраїчні атаки, кореляційні атаки, Strumok, Espresso, Krip.

ВСТУП

Впродовж останніх років спостерігається суттєве поширення сфери застосування потокових шифрів, що зумовлено розвитком інформаційних технологій, засобів передачі даних та помітним прогресом у дослідженні криптографічних властивостей алгоритмів потокового шифрування. Нині потокові шифри використовуються у вбудованих застосунках систем з обмеженою кількістю обчислювальних ресурсів, зокрема у бездротовій телефонії, в системах комутативного зв'язку та платного телебачення (більш докладну інформацію можна знайти, наприклад, в [1]). Окремо зазначимо шифр Strumok [2], який є національним стандартом потокового шифрування України [3].

Попри різноманіття наявних потокових шифрів, залишається актуальною задача створення апаратно-орієнтованих алгоритмів потокового шифрування, що відповідають підвищеним вимогам до швидкодії та мають прийнятну схемну складність. Як один з можливих підходів до розв'язання сформульованої задачі в цій статті запропоновано алгоритм потокового шифрування Krip, побудований на основі нелінійного регістру зсуву (НРЗ).

На відміну від інших потокових шифрів, побудованих на основі НРЗ, наприклад таких як Grain [4] та Espresso [5], в алгоритмі Krip використовується неавтономний регістр зсуву довжини 2 над алфавітом потужності 2^{256}, який функціонує аналогічно шифру Фейстеля з раундовою функцією, що використовується в алгоритмі шифрування Kalyna [6]. Це надає змогу певною мірою звести проблему стійкості запропонованого шифру стосовно низки атак до аналогічної проблеми стосовно зазначеного шифру Фейстеля.

Показано, що за стійкості на рівні 2^{256} операцій Krip забезпечує чотирикратний виграш у швидкодії порівняно з алгоритмом Strumok та майже двадцятикратний виграш порівняно з алгоритмом Espresso, хоча має суттєво більшу складність у разі апаратної реалізації, ніж останній.

Стаття має таку структуру. У розд. 1 наведено означення алгоритму Krip, у розд. 2 отримано результати аналізу його стійкості стосовно низки відомих атак, у розд. 3 наведено результати статистичного тестування шифрованих повідомлень, що отримуються за допомогою Krip, а в розд. 4 — відомості про характеристики його апаратної реалізації. Робота завершується стислими висновками.

1. ОПИС АЛГОРИТМУ ШИФРУВАННЯ

Для будь-якого натурального n позначимо V_n множину двійкових векторів довжини n. Для будь-яких цілих i, j покладемо $\overline{i, j} = \{k \in \mathbf{Z}: i \le k \le j\}$.

Запропонований алгоритм складається з двох процедур:

1) зашифрування/розшифрування повідомлень за допомогою НРЗ;

2) формування початкового стану НРЗ за ключем $k \in V_{256}$ та вектором ініціалізації $c \in V_{128}$.

Означення НРЗ. Нелінійний регістр зсуву — скінченний автомат з вхідним алфавітом $X = V_{256}$, вихідним алфавітом $Y = X$, множиною станів $S = V_{512}$ та функціями переходів і виходів, що визначаються відповідно за такими формулами:

$$h((s_1, s_2), x) = (s_2 \oplus \varphi(x \oplus s_1), s_1), \tag{1}$$

$$f((s_1, s_2), x) = x \oplus \psi_1(s_1, s_2) \oplus \psi_2(s_1, s_2), \tag{2}$$

де $x \in X$, $s_1, s_2 \in V_{256}$, φ та $\psi = (\psi_1, \psi_2)$ — підстановки на множинах V_{256} та V_{512} відповідно, $\psi_i: V_{512} \to V_{256}$, $i \in \overline{1,2}$.

Описаний НРЗ функціонує звичайним чином: якщо регістр перебуває у стані (s_1, s_2), де $s_1, s_2 \in V_{256}$, а на його вхід подається символ $x \in X$, то регістр виробляє вихідний символ $y = f((s_1, s_2), x)$ і переходить у наступний стан $h((s_1, s_2), x)$.

Зашифрування. Нехай $x_0, x_1, \ldots$ — відкритий текст, де $x_i \in X$, $i = 0, 1, \ldots$ ...,(s_1, s_0) — початковий стан НРЗ, $s_0, s_1 \in V_{256}$. Тоді знак y_i шифрованого тексту в i-му такті визначається за формулою

$$y_i = f((s_{i+1}, s_i), x_i), \tag{3}$$

де послідовність $s_0, s_1, \ldots$ задається за допомогою рекурентного співвідношення $(s_{i+2}, s_{i+1}) = h((s_{i+1}, s_i), x_i)$, яке можна записати у вигляді

$$s_{i+2} = s_i \oplus \varphi(x_i \oplus s_{i+1}), \ i = 0, 1, \ldots \tag{4}$$

Більш докладно: нехай (s_1, s_0) — початковий стан НРЗ, x_0 — знак відкритого тексту в такті $i = 0$. Тоді знак шифротексту в цьому такті обчислюється за формулою

$$y_0 = f((s_1, s_0), x_0) = x_0 \oplus \psi_1(s_1, s_0) \oplus \psi_2(s_1, s_0), \tag{5}$$

далі НРЗ переходить у наступний стан $(s_2, s_1) = h((s_1, s_0), x_0)$, де $s_2 = s_1 \oplus \varphi(x_0 \oplus s_0)$.

Якщо далі на вхід НРЗ подається знак відкритого тексту x_1, то на виході отримується знак шифротексту

$$y_1 = f((s_2, s_1), x_1) = x_1 \oplus \psi_1(s_2, s_1) \oplus \psi_2(s_2, s_1) \tag{6}$$

і регістр переходить у наступний стан $(s_3, s_2) = h((s_2, s_1), x_1)$, де $s_3 = s_2 \oplus \varphi(x_1 \oplus s_1)$. Далі процес зашифрування продовжується аналогічним чином.

$$M=\begin{pmatrix} x_{0,1} & x_{0,2} & x_{0,3} & x_{0,4} \\ x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} \\ x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} \\ x_{5,1} & x_{5,2} & x_{5,3} & x_{5,4} \\ x_{6,1} & x_{6,2} & x_{6,3} & x_{6,4} \\ x_{7,1} & x_{7,2} & x_{7,3} & x_{7,4} \end{pmatrix} \overset{L}{\mapsto} \begin{pmatrix} x_{0,1} & x_{0,2} & x_{0,3} & x_{0,4} \\ x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} \\ x_{2,2} & x_{2,3} & x_{2,4} & x_{2,1} \\ x_{3,2} & x_{3,3} & x_{3,4} & x_{3,1} \\ x_{4,3} & x_{4,4} & x_{4,1} & x_{4,2} \\ x_{5,3} & x_{5,4} & x_{5,1} & x_{5,2} \\ x_{6,4} & x_{6,1} & x_{6,2} & x_{6,3} \\ x_{7,4} & x_{7,1} & x_{7,2} & x_{7,3} \end{pmatrix}$$

Рис. 1. Перетворення L

Розшифрування. Нехай є відомими початковий стан (s_1, s_0) регістру та знак y_0 шифротексту в такті $i=0$. Тоді за формулою (5) можна знайти знак відкритого тексту x_0 і потім обчислити стан НРЗ $(s_2, s_1)=h((s_1, s_0), x_0)$ в такті $i=1$. Далі з використанням формули (6) за цим станом та знаком шифротексту y_1 можна знайти знак x_1 і потім обчислити стан НРЗ $(s_3, s_2)=h((s_2, s_1), x_1)$ у такті $i=2$. Взагалі, знаючи знак y_i та стан НРЗ (s_{i+1}, s_i) в i-му такті, можна обчислити знак x_i за формулою (3), після чого знайти стан НРЗ в $(i+1)$-му такті за формулою $(s_{i+2}, s_{i+1})=h((s_{i+1}, s_i), x_i)$, $i=0,1,\ldots$

Означення підстановки φ. Вона збігається з раундовою функцією, яка використовується у шифрі Kalyna з довжиною блоку 256 біт [6]. Зауважимо, що в цьому шифрі використовуються чотири підстановки: $\pi_0, \pi_1, \pi_2, \pi_3$ на множині V_8, на якій певним чином задано структуру поля $\mathbf{GF}(2^8)$, а також 8×8-матриця D над цим полем, що є максимально дистанційно роздільною [6, п. 5.3, 5.5].

Позначимо $\sigma_i=\pi_{i \bmod 4}$, $i\in\overline{0,7}$.

Нехай $u\in V_{256}$; тоді для обчислення значення $\varphi(u)$ використовується такий алгоритм:

1) запишемо вектор u у вигляді

$$u=(u_{0,1}, u_{1,1}, \ldots, u_{7,1}, u_{0,2}, u_{1,2}, \ldots, u_{7,2}, u_{0,3}, u_{1,3}, \ldots, u_{7,3}, u_{0,4}, u_{1,4}, \ldots, u_{7,4}),$$

де $u_{i,j}\in V_8$, та обчислимо вектор x з координатами $x_{i,j}=\sigma_i(u_{i,j})$, $i\in\overline{0,7}$, $j\in\overline{1,4}$, застосовуючи до кожної координати вектора u відповідну підстановку з набору $\pi_0, \pi_1, \pi_2, \pi_3$;

2) сформуємо з вектора x матрицю M розміру 8×4 та застосуємо до неї перетворення L (рис. 1);

3) помножимо кожен стовпець матриці $L(M)$ на матрицю D над полем $\mathbf{GF}(2^8)$ і отримаємо нову матрицю розміру 8×4 з елементами $v_{i,j}\in V_8$, $i\in\overline{0,7}$, $j\in\overline{1,4}$. Записуючи стовпці цієї матриці, починаючи з першого, один за одним у вигляді рядків, отримаємо вектор

$$v=(v_{0,1}, v_{1,1}, \ldots, v_{7,1}, v_{0,2}, v_{1,2}, \ldots, v_{7,2}, v_{0,3}, v_{1,3}, \ldots, v_{7,3}, v_{0,4}, v_{1,4}, \ldots, v_{7,4}),$$

який і дорівнює значенню $\varphi(u)$.

Означення підстановки ψ. Ця підстановка побудована на основі раундової функції, що використовується у шифрі Kalyna з довжиною блоку 512 біт.

Точніше: нехай $u\in V_{512}$; тоді для обчислення значення $\psi(u)$ використовується такий алгоритм:

1) запишемо вектор u у вигляді

$$u=(u_{0,1}, u_{1,1}, \ldots, u_{7,1}, u_{0,2}, u_{1,2}, \ldots, u_{7,2}, \ldots, u_{0,8}, u_{1,8}, \ldots, u_{7,8}), \qquad (7)$$

де $u_{i,j}\in V_8$, та обчислимо вектор x з координатами $x_{i,j}=\sigma_i(u_{i,j})$, $i\in\overline{0,7}$, $j\in\overline{1,8}$, застосовуючи до кожної координати вектора u відповідну підстановку з набору $\pi_0, \pi_1, \pi_2, \pi_3$;

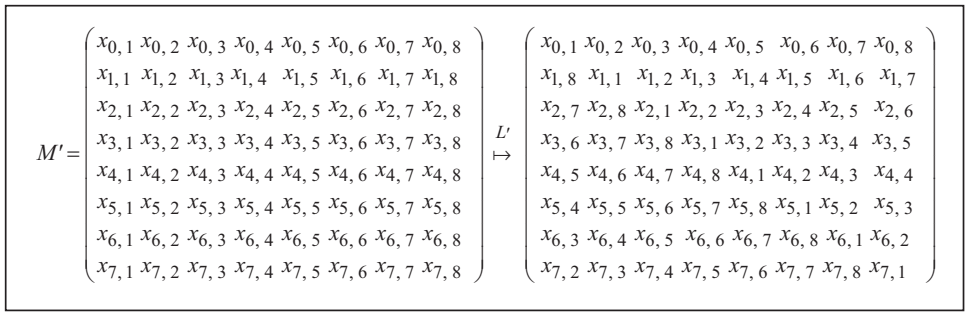

Рис. 2. Перетворення L'

2) сформуємо з вектора x матрицю M' розміру 8×8 та застосуємо до неї перетворення L', як показано на рис. 2;

3) помножимо кожен стовпець матриці $L'(M')$ на матрицю D над полем $\mathbf{GF}(2^8)$, у результаті отримаємо нову матрицю розміру 8×8 з елементами $v_{i,j} \in V_8$, $i \in \overline{0,7}$, $j \in \overline{1,8}$. Запишемо стовпці цієї матриці, починаючи з першого, один за одним у вигляді рядків і отримаємо вектор

$$v = (v_{0,1}, v_{1,1}, ..., v_{7,1}, v_{0,2}, v_{1,2}, ..., v_{7,2}, ..., v_{0,8}, v_{1,8}, ..., v_{7,8});$$

4) застосовуючи до кожної координати вектора v відповідну підстановку з набору $\pi_0, \pi_1, \pi_2, \pi_3$, визначимо значення $\psi(u)$ як вектор з координатами $\sigma_i(v_{i,j})$, $i \in \overline{0,7}$, $j \in \overline{1,8}$.

Означення процедури формування початкового стану НРЗ. Процес формування початкового стану НРЗ визначається на основі масштабованої схеми блокового шифру Camellia [7]. Необхідність масштабування зумовлена тим, що у шифрі Camellia довжини блоку та ключа дорівнюють відповідно 128 та 256 біт, тоді як в алгоритмі Кгір аналогічні значення параметрів складають відповідно 512 та 256 біт.

Константи C_1 , …, C_6, що використовуються у зазначеній процедурі, визначаються за наведеним нижче алгоритмом, подібним до алгоритму вибору констант $\Sigma_{i(64)}$, $i \in \overline{1,6}$, у розкладі ключів шифру Camellia [7, п. 3.4, табл. 1].

Викладемо алгоритм формування констант C_i, $i \in \overline{1,6}$.

1. Обчислюється значення $\sqrt{p_i}$, де p_i — i-те у порядку зростання просте число, результат подається у двійковій системі числення з точністю 261 двійковий знак після коми. Ці 261 двійкові знаки записуються у вигляді двійкового вектора $c_i = (c_{i,1}, \ldots, c_{i,261})$.

Таблиця 1. Значення, що використовуються для формування початкового стану НРЗ

Константи	Значення констант
C_1	A09E667F 3BCC908B 2FB1366E A957D3E3 ADEC1751 2775099D A2F590B0 667322A9
C_2	B67AE858 4CAA73B2 5742D707 8B83B892 5D834CC5 3DA4798C 720A6486 E45A6E24
C_3	C6EF372F E94F82BE 73980C0B 9DB90682 1044ED7E 744E4A3F 0D8D423A 1831D2A4
C_4	54FF53A5 F1D36F1C EA7E61FC 37A20D54 A77FE7B7 8415DFC8 E34A6FE8 E2DF92A4
C_5	10E527FA DE682D1D E49E330E 42B4CBB2 9BA5A455 316E0C65 507CD18E 9E51E694
C_6	B05688C2 B3E6C1FD BD99E6FF 3C90BDC4 DBC64712 A5BB1687 67E27C3C F76C8E72

2. Із вектора c_i формується вектор $C_i = (c_{i,5}, \ldots, c_{i,260})$, отриманий вилученням перших чотирьох та останньої координат.

3. Вектор $C_i = (c_{i,5}, \ldots, c_{i,260})$ поділяється на тетради, кожна з яких перетворюється на відповідний символ шістнадцяткової системи числення. Отриманий таким чином вектор, що складається з 64 шістнадцяткових символів, і є

шістнадцятковим поданням константи C_i. (Зазначимо, що в процесі формування початкового стану НРЗ використовується відповідне двійкове подання константи C_i.)

Перелік констант C_i у шістнадцятковому поданні наведено в табл. 1.

Процес формування початкового стану НРЗ за ключем $k \in V_{256}$ та за вектором ініціалізації $c \in V_{128}$ має такий вигляд.

1. Запишемо у регістр вектор $(k_1 \oplus \bar{c}, k_1 \oplus k_2 \oplus c, k_1 \oplus k_2 \oplus \bar{c}, k_2 \oplus c)$, де $k = (k_1, k_2)$, $k_1, k_2 \in V_{128}$, а $\bar{c}$ позначає вектор з координатами, інвертованими до координат вектора c. Позначимо $K_L = (k_1 \oplus \bar{c}, k_1 \oplus k_2 \oplus c)$, $K_R = (k_1 \oplus k_2 \oplus \bar{c}, k_2 \oplus \bar{c})$.

2. До вектора (K_L, K_R) двічі застосуємо функцію (1), використовуючи константи C_1 та C_2; у результаті отримаємо вектор $(U_1, U_2) = h(h((K_L, K_R), C_1), C_2)$.

3. Обчислимо вектор $(U_3, U_4) = (U_1, U_2) \oplus ((K_L, K_R) <<<_{128})$.

4. До вектора (U_3, U_4) застосуємо двічі функцію (1), використовуючи константи C_3 та C_4; результатом буде вектор $(K_{AL}, K_{AR}) = h(h((U_3, U_4), C_3), C_4)$.

5. Обчислимо $(U_5, U_6) = (K_{AL}, K_{AR}) \oplus ((U_3, U_4) <<<_{128})$.

6. До вектора (U_5, U_6) застосуємо двічі функцію (1), використовуючи константи C_5 та C_6; результатом буде вектор $(K_{BL}, K_{BR}) = h(h((U_5, U_6), C_5), C_6)$.

7. Обчислимо вектори $K_1, \ldots, K_{32}$, як зазначено нижче:

$K_1 = K_L$,
$K_2 = K_R$,
$K_3 = K_{BL}$,
$K_4 = K_{BR}$,
$K_5 = K_{AL} <<<_{15}$,
$K_6 = K_{AR} <<<_{15}$,
$K_7 = K_L \oplus (K_{BL} <<<_{15})$,
$K_8 = K_R \oplus (K_{BR} <<<_{15})$,
$K_9 = K_{AL} <<<_{30}$,
$K_{10} = K_{AR} <<<_{30}$,
$K_{11} = K_{BL} <<<_{30}$,
$K_{12} = K_{BR} <<<_{30}$,
$K_{13} = K_L \oplus (K_{AL} <<<_{45})$,
$K_{14} = K_R \oplus (K_{AR} <<<_{45})$,
$K_{15} = K_{BL} <<<_{45}$,
$K_{16} = K_{BR} <<<_{45}$,
$K_{17} = K_{AL} <<<_{60}$,
$K_{18} = K_{AR} <<<_{60}$,
$K_{19} = K_L <<<_{60}$,
$K_{20} = K_R <<<_{60}$,
$K_{21} = K_{AL} \oplus (K_{BL} <<<_{77})$,
$K_{22} = K_{AR} \oplus (K_{BR} <<<_{77})$,
$K_{23} = K_{BL} <<<_{77}$,
$K_{24} = K_{BR} <<<_{77}$,
$K_{25} = K_{AL} <<<_{94}$,
$K_{26} = K_{AR} <<<_{94}$,
$K_{27} = K_L \oplus (K_R <<<_{94})$,
$K_{28} = K_R \oplus (K_{AL} <<<_{94})$,
$K_{29} = K_{AL} \oplus (K_{AR} <<<_{111})$,
$K_{30} = K_{AR} \oplus (K_{AL} <<<_{111})$,
$K_{31} = K_{BL} \oplus (K_{BR} <<<_{111})$,
$K_{32} = K_{BR} \oplus (K_{BL} <<<_{111})$.

8. Обчислимо вектор $(K_1^*, K_2^*) = h(h \ldots h(h((K_L, K_R), K_1), K_2) \ldots, K_{32})$, застосовуючи 32 рази до вхідного вектора (K_L, K_R) функцію (1) та використовуючи як вхідні символи вектори $K_1, \ldots, K_{32}$, обчислені в п. 7. Отриманий вектор (K_1^*, K_2^*) є результатом визначеної процедури.

2. АНАЛІЗ СТІЙКОСТІ

Алгебраїчні атаки. Нехай є відомими відкритий текст $x_0, x_1, \ldots$ та відповідний йому шифрований текст $y_0, y_1, \ldots$, який отримано за початковим станом (s_1, s_0) НРЗ згідно з формулами (3), (4). Тоді на підставі формул (1), (2) для відновлення початкового стану можна скласти систему рівнянь

$$\psi_1(s_{i+1}, s_i) \oplus \psi_2(s_{i+1}, s_i) = x_i \oplus y_i, \tag{8}$$
$$(s_{i+1}, s_i) = (h_{x_i} \circ \cdots \circ h_{x_0})((s_1, s_0)), \ i = 0, 1, \ldots,$$

де символ ∘ позначає операцію композиції відображень, $h_x(s', s'') = h((s', s''), x)$ для будь-яких $s', s'' \in V_{256}$, $x \in X$.

Система (8) містить 512 змінних — координат вектора (s_1, s_0), проте її можна замінити рівносильною системою рівнянь від 256 змінних. Для цього слід скористатися означенням підстановки ψ, а також тим, що довільна рівність вигляду $\psi_1(u) \oplus \psi_2(u) = w$ є рівносильною рівності, яка виражає певні 256 координат вектора u в його представленні (6) через інші 256 координат цього вектора (цей вираз задається нелінійною функцією, яка залежить від підстановок $\pi_0, \pi_1, \pi_2, \pi_3$, матриці D та перетворення L', наведеного на рис. 2).

Зауважимо, що алгебраїчна імунність кожної підстановки $\pi_0, \pi_1, \pi_2, \pi_3$ дорівнює 3 [8]. Спроби застосувати до отриманої системи рівнянь відомі методи розв'язання, ефективніші за повний перебір, мають принципові труднощі, пов'язані з аналітичною складністю рівнянь у системі та непередбачуваністю їхніх числових характеристик (зокрема, алгебраїчного степеня). Отже, нині стійкість алгоритму Krip до алгебраїчних атак оцінюється на рівні 2^{256}.

Атаки, що базуються на гомоморфізмах. Якщо існує гомоморфізм НРЗ в автомат з тими самими вхідним та вихідним алфавітами з меншою кількістю станів, то до описаного алгоритму шифрування можна застосувати атаки, що базуються на гомоморфізмах [9, 10]. Достатньою умовою неможливості застосування таких атак є примітивність групи, яка породжена підстановками h_x, де $x \in X$, а функція h визначається за формулою (1). У роботі [11] отримано критерій примітивності групи підстановок зазначеного вигляду, з якого випливає, що ця група є примітивною, якщо група, породжена підстановкою φ та усіма підстановками вигляду $x \mapsto x \oplus k$, де $x, k \in V_{256}$, є примітивною. Проте згідно з твердженням 4 з роботи [12] остання група є знакозмінною на множині V_{256}, а отже й примітивною, що гарантує стійкість алгоритму Krip до атак на основі гомоморфізмів.

Кореляційні атаки. Оскільки Krip не містить лінійних компонент, то класичні кореляційні атаки на фільтрувальні чи комбінувальні генератори гами, побудовані на основі лінійних регістрів зсуву [13], є незастосовними до нього. Із системи рівнянь (8) випливає, що існує близький зв'язок між задачею знаходження лінійних наближень перетворень, які реалізуються за допомогою алгоритму Krip, та задачею побудови лінійних атак на шифр Фейстеля з раундовою функцією φ, яка використовується в алгоритмі шифрування Kalyna з довжиною блоку 256 біт. На сьогодні подібні (ефективні) атаки невідомі.

Зауважимо також, що за допомогою наслідку 2 в [14] і даних, наведених у табл. 2 з роботи [12], неважко переконатися, що нелінійність функції $F(s) = \psi_1(s) \oplus \psi_2(s)$, $s \in V_{512}$, яка визначається за формулою $\mathrm{nl}(F) = 2^{511} - 1/2 \cdot \max\limits_{\substack{v \in V_{256} \setminus \{0\}, \\ u \in V_{512}}} \left| \sum\limits_{s \in V_{512}} (-1)^{vF(s) \oplus us} \right|$, є не меншою за $2^{511}(1 - 9^4 \cdot 2^{-32})$.

Структурні властивості внутрішніх послідовностей НРЗ. Розглянемо послідовність станів $s_0, s_1, \ldots, s_t$, яка виробляється за початковим станом $s_0 \in S$ та вхідною послідовністю $x_0, x_1, \ldots, x_{t-1}$ регістру за законом $s_{i+1} = h(s_i, x_i)$, $i \in \overline{0, t-1}$.

Твердження 1. Нехай елемент s_0 та послідовність $x_0, x_1, \ldots, x_{t-1}$ вибираються незалежно випадково та рівноймовірно з множин S та X^t відповідно. Тоді ймовірність того, що всі стани $s_0, s_1, \ldots, s_t$ є попарно різними, обмежена знизу величиною $1 - \dfrac{t(t-1)}{2|S|}$.

Доведення. Розглянемо позначений орієнтований граф на множині вершин S, в якому з вершини s у вершину $\widetilde{s}$ спрямована дуга, позначена символом $x \in X$, тоді й тільки тоді, коли $h(s, x) = \widetilde{s}$. Неважко переконатися в тому, що для довільних елементів $s, \widetilde{s} \in S$ та будь-якого натурального $l \geq 2$ в цьому графі існує точно $|X|^{l-2}$ шляхів, що напрямлені з вершини s у вершину $\widetilde{s}$. Звідси випливає, що для будь-яких цілих $0 \leq i < j \leq t-1$ та довільного $s = (s', s'') \in S$ ймовірність події $\{s_i = s_j = s\}$ визначається такою рівністю:

$$\mathbf{P}\{s_i = s_j = s\} = \begin{cases} |X|^{-4}, \text{ якщо } j - i \geq 2; \\ |X|^{-3}, \text{ якщо } j - i = 1 \text{ та } s' = s''; \\ 0 \qquad \text{у протилежному випадку.} \end{cases}$$

Отже, ймовірність того, що випадкова послідовність $s_0, s_1, \ldots, s_t$ містить принаймні два однакових елементи, дорівнює

$$\mathbf{P}\left(\bigcup_{\substack{0 \leq i < j \leq t-1, \\ s \in S}} \{s_i = s_j = s\} \right) \leq \sum_{\substack{0 \leq i < j \leq t-1: \, j-i \geq 2, \\ s \in S}} \mathbf{P}\{s_i = s_j = s\} + \sum_{\substack{0 \leq i < j \leq t-1: \, j-i = 1, \\ s \in S}} \mathbf{P}\{s_i = s_j = s\} =$$

$$= \sum_{0 \leq i < j \leq t-1: \, j-i \geq 2} |S| \cdot |X|^{-4} + \sum_{0 \leq i < j \leq t-1: \, j-i = 1} |X| \cdot |X|^{-3} \leq \frac{t(t-1)}{2|X|^2},$$

що й треба було довести.

Зазначене твердження засвідчує, що для послідовності станів НРЗ, яка отримується за випадково обраними (незалежними рівноймовірними) початковим станом та вхідною послідовністю, виконується так званий «парадокс днів народження» так само, як і для суто випадкової послідовності елементів множини S. Зокрема, у послідовності станів НРЗ з високою ймовірністю нема повторень, якщо t є помітно меншим за $\sqrt{|S|} = 2^{256}$.

Отже, відносна частка таких пар (початковий стан, вхідна послідовність довжини t), для яких послідовність станів містить фрагмент, що повторюється, є нехтовно малою, якщо t є суттєво меншим за 2^{256}.

3. РЕЗУЛЬТАТИ СТАТИСТИЧНИХ ДОСЛІДЖЕНЬ

Статистичні дослідження алгоритму шифрування виконувались за такими напрямками.

1. Перевірка статистичної якості алгоритму як генератора псевдовипадкових послідовностей.

2. Перевірка незалежності внутрішніх послідовностей, отриманих для того ж самого ключа, різних векторів ініціалізації та нульової послідовності як відкритого тексту.

Опишемо докладніше ці дослідження.

Перевірка статистичної якості алгоритму як генератора псевдовипадкових послідовностей. Перевірка виконувалася за методикою, визначеною в [15]: кожен з 16 тестів пакету NIST застосовуваний із рівнем значущості $\alpha = 0.01$ до n шифрованих повідомлень довжиною 10^6 біт, отриманих за нульових відкритих повідомлень для (випадково згенерованого) ключа

3C4920A999A32DFE892761617E5FED65E5B5F5E41E7D97FC4993E05EFC8BA06B

та 1000 різних випадкових векторів ініціалізації, де $n = 6233$ для тестів RandomExcursion та RandomExcursionVariant (така кількість послідовностей

вибрана з певних технічних причин, пов'язаних з особливостями цих тестів) та $n = 10000$ для всіх інших тестів.

Для кожного тесту виконано такі операції:

1) отримано n P-величин $P_1, ..., P_n$ (означення P-величини наведено в [15, с. 5]);

2) обчислено значення $\chi^2 = \sum_{i=1}^{10} \frac{(F_i - n/10)^2}{n/10}$, де n — кількість послідовностей в статистичному експерименті, F_i — кількість P-величин, що належать інтервалу $\left(\frac{i-1}{10}, \frac{i}{10}\right)$, $i \in \overline{1,10}$;

3) обчислено величину $P_T = \text{igams}(9/2, \chi^2/2)$, де igams — неповна гама-функція, що визначається за формулою

$$\text{igams}(a, x) = \frac{1}{\Gamma(a)} \int_x^\infty t^{a-1} e^{-t} dt \ , \text{ де } \Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt.$$

Згідно з [15] генератор, що розглядається, проходить тест, якщо для нього виконуються такі дві умови:

$$1 - 0.01 - 3\sqrt{\frac{0.01(1-0.01)}{n}} \le \frac{k}{n} \le 1 - 0.01 + 3\sqrt{\frac{0.01(1-0.01)}{n}}, \qquad (9)$$

де k — кількість послідовностей, що пройшли тест; та

$$P_T \ge 0.0001. \qquad (10)$$

Для алгоритму шифрування Krip умови (9) та (10) виконуються для усіх 16 тестів. Для тестів RandomExcursion та RandomExcursionVariant формула (9) визначає інтервал $(0.98622,\ 0.99378)$; для всіх інших тестів — інтервал $(0.98702,\ 0.99298)$. Значення пропорції k/n для тестів потрапляє у визначені інтервали (причому для тестів RandomExcursion та RandomExcursionVariant збігається з лівою границею інтервалу). Отже, згідно з [15] Krip можна вважати генератором псевдовипадкових послідовностей.

Перевірка незалежності внутрішніх послідовностей НРЗ. Для виконання цього статистичного дослідження застосовувались результати [16] стосовно перевірки попарної незалежності довільної кількості випадкових послідовностей з використанням вибіркової кореляційної матриці.

Всього виконано 20 експериментів, в кожному з яких для випадково згенерованого ключа формувалося $m = 100$ внутрішніх послідовностей довжиною $l = 10^6$ біт кожна, отриманих для нульової вхідної послідовності та 100 випадково вибраних векторів ініціалізації $IV_1, \ldots, IV_{100}$.

Кожен експеримент складається з таких кроків:

1) сформувати бітову послідовність $\Xi^{(j)} = (\xi_1^{(j)}, \ldots, \xi_l^{(j)})$, отриману з внутрішньої послідовності, яка відповідає вектору IV_j, $j \in \overline{1,100}$;

2) за отриманими послідовностями визначити величини

$$D_{ij} = \frac{1}{l-1} \sum_{k=1}^{l} (\xi_k^{(i)} - 0.5)(\xi_k^{(j)} - 0.5),$$

що є вибірковими попарними коваріаціями, та побудувати вибіркову кореляційну матрицю

$$\mathbf{R} = (R_{ij})_{i,\,j=1}^{100}, \text{ де } R_{ij} = \frac{D_{ij}}{\sqrt{D_{ii}}\sqrt{D_{jj}}}, \ i, j \in \overline{1,100}.$$

Зауважимо, що для попарно незалежних випадкових послідовностей $\Xi^{(j)}$, $j \in \overline{1,100}$, матриця $\mathbf{R}$ збігається з одиничною матрицею I, отже її визначник дорівнює 1. Але в нашому випадку ця матриця задається як емпірична за результатами статистичних експериментів. Тому для побудови критерію

Таблиця 2. Значення детермінанта кореляційної матриці, отримані за результатами 20 проведених експериментів

Номер ключа	Значення ключа	Значення детермінанта
1	3C4920A999A32DFE892761617E5FED65 E5B5F5E41E7D97FC4993E05EFC8BA06B	0.995105
2	E87D2A7AB61AE3C1D110E2411DDBBBC4 5C8D1564907631DBC454B6BC336E6EFC	0.995111
3	BD16E669884A36BF40E1BA5D9A7966AC A9D51B6EF44D3861A3EC93396AF829A0	0.995235
4	3F15DEF66809AC3551BEDC15BF1616C9 8FCDBFD881E7A5FB17867A73AE26EB3	0.995101
5	57F5C606C18C91B028496B4C8832875E B8C8937D1EF36C29CBB624B13CE7B5CC	0.995236
6	B0D4F772B4D8668266DA794C3F187A7D CB1BBDA921026AAE20E6F5E885B51BEC	0.995081
7	30A8DBDE96874904CA3E238D38ED2FC5 91CB7124D68925CFF6E9E75782E5CA56	0.995031
8	D031E8881C34BDD6E63C68BCFF0E96FC F7F8F7E56BCDCB5A5630AA287F2CD8DF	0.995208
9	AE2E3E05DB0D5EC0FEC779C46DDD6F30 95BF47B274FCDE67D508EBB0675752B	0.995162
10	43A5F2C4B6774DA5EDD3CE4DC47AAB6B 6893ED09B8D05B557DEAD91DD1B1C210	0.995138
11	7B533721E7615EFFE6D04EC528232A8E 705E5591EA286DA95312E7E57B311F7	0.995102
12	D816A9527C96981CE33E2681FF0855B BFB329EB282FCEE93F8F396171A5660E	0.995204
13	84E3C5A5B8F5B20385B1FB5556AA1880 675482745E7DE9C73A92CC45A78420C	0.995168
14	9EA55C5369DA43956B8897B0136DFEE6 86FDC7856220199EF9E99A0B6566D32	0.995280
15	40B2598A529AB9DC9C84D3222572D1CA 9BE02624299B29B99F2431ABB12EC3EB	0.995141
16	A257D64FCA7D62549EAFAB8B6BAA3F54 CA766D66BA2BFA8FC5B7964D42191D25	0.995123
17	EAA1B8B45F171CB6FED976239DB91194 598B4519B682298DEABC6BBA1F2F5D28	0.995262
18	E99A9E14E8B4710BB852708118303EA 6A828539830EC4E082D98D6787E7D3F7	0.995168
19	20CEE16EE94997247064103DBFC9FB6 32E321F5B578269EF2069E41D1B36A5	0.995093
20	E4A1CC943D0C6D7EF191473274E343C C2FFC014A95D8003FD4FC8BB1B7FC8D	0.995058
	Вибіркове математичне сподівання	0.99515035

перевірки гіпотези H_0, що «послідовності є попарно незалежними» з рівнем значущості $\alpha_0 = 10^{-3}$, використовуватимемо визначник матриці $\mathbf{R}$: $d = \det \mathbf{R}$.

Закон розподілу визначника d досить складний, але для достатньо великих значень l можна використовувати його асимптотичне подання: якщо послідовності є попарно незалежними, то

$$\mathbf{P}\{-nd \le v\} = \mathbf{P}\{\chi^2_f \le v\} + \frac{\gamma}{n^2}(\mathbf{P}\{\chi^2_{f+4} \le v\} - \mathbf{P}\{\chi^2_f \le v\}) + \mathrm{O}(n^{-3}),$$

Таблиця 3. Значення детермінанта кореляційної матриці для 20 серій зі 100 послідовностей довжини 1000000 біт, згенерованих сертифікованим фізичним генератором випадкових послідовностей

Номер серії	Значення детермінанта	Номер серії	Значення детермінанта
1	0.995169	11	0.995223
2	0.995150	12	0.995219
3	0.995164	13	0.995193
4	0.995178	14	0.995174
5	0.995172	15	0.995205
6	0.995182	16	0.995188
7	0.995163	17	0.995219
8	0.995168	18	0.995247
9	0.995166	19	0.995227
10	0.995201	20	0.995205
Вибіркове математичне сподівання			0.99519065

де $f=\dfrac{m(m-1)}{2}$, $n=l-\dfrac{2m+11}{6}$, $\gamma=\dfrac{m(m-1)(2m^2-2m-13)}{288}$, χ^2_f — випадкова величина, що має χ-квадрат розподіл з f степенями свободи.

Зазначимо, що для $l=10^6$, $m=100$ можна використовувати наближену формулу

$$\mathbf{P}\{-nd\le v\}\approx\mathbf{P}\{\chi^2_f\le v\}. \tag{11}$$

Оскільки у разі наближення вилучаються додатні величини у правій частині (11), то отримана критична область відповідає більшому рівню значущості, ніж заданий.

Виконавши відповідні обчислення з використанням (11), отримаємо критичну область для параметра $d=\det R$, що відповідає рівню значущості $\alpha_0=10^{-3}$: $d\in(0,e^{-1})$.

За результатами проведених експериментів здобуті значення d (див. табл. 2), які засвідчують на користь гіпотези H_0 про попарну незалежність внутрішніх послідовностей НРЗ. Для порівняння в табл. 3 наведено величини d для 20 серій зі 100 послідовностей з 10^6 випадковими бітами в кожній.

4. ПОРІВНЯННЯ ХАРАКТЕРИСТИК АПАРАТНИХ РЕАЛИЗАЦІЙ АЛГОРИТМІВ ПОТОКОВОГО ШИФРУВАННЯ STRUMOK, KRIP ТА ESPRESSO

У табл. 4 наведено характеристики проєктів апаратних реалізацій алгоритмів Strumok та Krip у середовищі проєктування Quartus Prime 17.1 Lite Edition, а також відповідні характеристики апаратної реалізації шифру Espresso.

Зауважимо, що для мінімально можливої кількості тактів (для цього випадку 2) апаратна реалізація алгоритму Strumok зашифровує слова довжиною 64 біт, а реалізація алгоритму Espresso — 1 біт; водночас реалізація алгоритму Krip зашифровує слова довжиною 256 біт. Отже, отримуємо відповідно чотирикратний та двадцятикратний виграш у швидкодії для алгоритму Krip. До того ж слід зауважити, що зазначений виграш досягається за рахунок значно більших апаратних затрат. Зокрема, в алгоритмі Strumok реалізовано 64-розрядний елемент раундової функції алгоритму Kalyna, в той час як алгоритм Krip містить дві раундові функції, які використовуються у версіях алгоритму Kalyna з довжиною блоку 256 та 512 біт відповідно, а алгоритм Espresso взагалі не використовує табличних елементів.

Таблиця 4. Характеристики апаратних реалізацій алгоритмів Strumok, Krip та Espresso

Назва параметра	Значення параметрів для алгоритмів		
	Espresso	**Strumok**	**Krip**
Довжина слова, біт	1	64	256
Тактова частота, МГц	2217	350	350
Кількість тактів перетворення	1	2	2
Кількість логічних блоків	1497	3100	8015
Кількість блоків пам'яті 256x64	–	64	768
Кількість блоків пам'яті 256x8	–	8	64
Швидкість шифрування, Гбіт/с	2.22	11.2	44.8

Основний ресурс, необхідний для збільшення швидкодії у разі реалізації цих двох раундових функцій, це пам'ять, яка потрібна для зберігання таблиць. Сучасні інтегральні схеми FPGA містять значну кількість блоків пам'яті, достатню для реалізації алгоритму Krip за помірної вартості апаратури. Наприклад, використання FPGA Arria V GT 5AGTC3 фірми Intel з 1051 блоком пам'яті та чотирма трансіверами на 10.3125 Гбіт/с дає змогу виконувати шифрування даних у мережах зі швидкістю, більшою за 40 Гбіт/с.

ВИСНОВКИ

У статті запропоновано апаратно-орієнтований алгоритм потокового шифрування Krip. Цей алгоритм побудовано на основі неавтономного нелінійного регістру зсуву довжини 2 над алфавітом потужності 2^{256} та є високошвидкісним потоковим шифром з рівнем стійкості, який відповідає сучасним вимогам. У подальшому заплановано втілення алгоритму Krip у серію високошвидкісних засобів шифрування для криптографічного захисту інформації, що циркулює мережею Ethernet між Центрами оброблення даних. Вважається, що така апаратура забезпечить шифрування на швидкості лінії (тобто без втрати пакетів) з низькими привнесеними затримками не тільки у топологіях «точка–точка», але і у інших багатоточкових топологіях.

Реалізація алгоритму Krip дає змогу виконувати шифрування даних на рівні L2 згідно з моделлю ISO у територіальних мережах (таких як Metro Ethernet) та в опорній мережі оператора зв'язку (Carrier Ethernet) без деградації параметрів високошвидкісних оптичних каналів зв'язку, побудованих на обладнанні Cisco NexusN5K-C5672UP (близько шести 40G QSFP).

СПИСОК ЛІТЕРАТУРИ

1. Сторожук А.Ю. Методи оцінювання та обґрунтування стійкості потокових шифрів відносно статистичних атак на основі алгебраїчно вироджених наближень булевих функцій. Дис. канд. техн. наук: 21.05.01. Київ, 2016. 176 с.
2. Gorbenko I., Kuznetsov A., Gorbenko Yu., Alekseychuk A., Timchenko V. Strumok Keystream Generator. The 9th IEEE International Conference on Dependable Systems, Services and Technologies, DESSERT'2018, 24–27 May, 2018, Kyiv, Ukraine. P. 292–299.
3. ДСТУ 8845:2019 Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного потокового перетворення. Київ: ДП «УкрНДНЦ», 2019.
4. Hell M., Johansson T., Maximov A., Meier W. The Grain family of stream cipher. New Stream Ciher Design: The eSTREAM Finalists. LNCS 4986. 2008. P. 179–190.

5. Dubrova E., Hell M. Espresso: A stream cipher for 5G wireless communication systems. Cryptology ePrint Archive. URL: http://eprint.iacr.org/2015/241.

6. Oliynykov R., Gorbenko I., Kazymyrov O., Ruzhentsev V., Kuznetsov O., Gorbenko Yu., Dyrda O., Dolgov V., Pushkaryov A., Mordvinov R., Kaidalov D. A new encryption standard of Ukraine: The Kalyna Block Cipher. Cryptology ePrint Archive. URL: http://eprint.iacr.org/2015/650.

7. Aoki A., Ichikawa T., Kanda M., Matsui M., Moriai S., Nakajima J., Tokita T. Camellia: a 128-bit block cipher suitable for multiple platforms — Design and Analysis. Selected Areas in Cryptography — SAC 2001. Proceedings: Springer Verlag, 2001. P. 39–56.

8. Олійников Р.В. Горбенко І.Д., Казимиров О.В. Принципи побудови і основні властивості нового національного стандарту блокового шифрування України. *Захист інформації*. 2015. Т. 17, № 2. С. 142–157.

9. Шапошников И.Г. О конгруэнциях конечных многоосновных универсальных алгебр. *Дискретная математика.* 1999. Т. 11, Вып. 3. С. 48–62.

10. Горчинский Ю.Н. О гомоморфизмах многоосновных универсальных алгебр в связи с криптографическими применениями. Труды по дискретной математике. Т. 1. Москва: ТВП, 1997. С. 67–84.

11. Алексейчук А.Н., Скрыпник Л.В. Критерий примитивности группы подстановок, порожденной раундовыми преобразованиями шифра Фейстеля. *Радиотехника*. 2005. Вып. 141. С. 31–39.

12. Алексейчук А.Н., Ковальчук Л.В., Шевцов А.С., Яковлев С.В. О криптографических свойствах нового национального стандарта шифрования Украины. *Кибернетика и системный анализ*. 2016. Т. 52, № 3. С. 16–32.

13. Meier W. Fast correlation attacks: Methods and countermeasures. Fast Software Encryption. FSE'2011. Proceedings: Springer Verlag, 2011. P. 55–67.

14. Park S., Sung J., Lee S., Lim J. Improving the upper bound on the maximum differential and the maximum linear hull probability for the SPN structures and AES. Fast Software Encryption. FSE'03. Proceedings: Springer Verlag, 2003. P. 247–260.

15. A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST Special Publication 800-22, 1999. Rev. 1. 131 p.

16. Андерсон Т. Введение в многомерный статистический анализ. Москва; Ленинград: Физматгиз, 1963. 500 с.

L.V. Kovalchuk, I.V. Koriakov, A.N. Alekseychuk

KRIP: HIGH-SPEED HARDWARE-ORIENTED STREAM CIPHER BASED ON NON-AUTONOMOUS NON-LINEAR SHIFT REGISTER

Absract. A stream cipher based on a non-autonomous non-linear shift register of length 2 over the alphabet of 2^{256} symbols is proposed. This register works like a Feistel cipher with a round function, used in cipher Kalyna. It is shown that under the security level 2^{256} the cipher Krip is four times faster than the current National Encryption Ukrainian Standard and is almost 20 times faster than the modern stream cipher Espresso.

Keywords: stream cipher, Feistel scheme, non-linear shift register, generator of pseudorandom sequences, algebraic attacks, correlation attacks, Strumok, Espresso, Krip.

Надійшла до редакції 26.07.2022