Харьковский национальный университет радиоэлектроники

Международная академия наук прикладной радиоэлектроники

ПРИКЛАДНАЯ РАДИОЭЛЕКТРОНИКА

Научно-технический журнал

Адрес редакции:
Редакция журнала «Прикладная радиоэлектроника»
Харьковский национальный университет радиоэлектроники
просп. Ленина, 14, 61166, Харьков, Украина
Тел.: + 38 (057) 702 10 57
Факс: + 38 (057) 702 10 13
E-mail: are@kture.kharkov.ua
http://www.anpre.org.ua

СОДЕРЖАНИЕ

УВАЖАЕМЫЕ ЧИТАТЕЛИ!

Выпуск настоящего журнала является тематическим и посвящен проблемным вопросам криптологии и ряду аспектов защиты информации. Статьи в основном являются заказными и подготовлены специалистами по соответствующим направлениям. Большинство статей посвящено новым и уже ставшим традиционными криптографическим преобразованиям. Однако, при заказе статей мы руководствовались в основном практическими аспектами криптологии, ориентируясь на задачи, которые решаются нашим спонсором – ПАТ «Институт информационных технологий».

В первом разделе журнала представлены статьи, которые посвящены теории и практике симметричных криптографических преобразований, в основном блочным симметричным шифрам (БСШ). По-прежнему актуальными являются исследования, связанные с решением систем линейных булевых уравнений с искаженными частями. На наш взгляд, интересные результаты в этом направлении получены профессором Алексейчуком А.Н, которые представлены в первой статье журнала. В статье профессора Долгова В.И. и доцента Лисицкой И.В. излагается уточнённый подход к определению Марковских шифров, основывающийся на стохастических уравнениях Марковских процессов. Такой подход является дискуссионным и требует обсуждения. Определенное внимание заслуживает статья, подготовленная под руководством профессора Долгова В.И., посвященная вопросам практической оценки максимальных значений дифференциалов и линейных корпусов марковских шифров. Также в связи с признанием направления, связанного с развитием БСШ на основе применения схемы Лея – Массея, представлены исследования по оценке сложности различения схемы Лея – Массея и случайной1 перестановки, авторы доцент Олейников Р.В. и аспирант Кайдалов Д.С. Статья доцента Руженцева В.И. посвящена исследованию особенностей организации интегральных атак на различные варианты шифров с rijndael-подобными преобразованиями. Существенного внимания заслуживает и статья доцента Халимова Г.З. и аспиранта Бойко А.А., в которой предлагается метод универсального хеширования по рациональным функциям алгебраических кривых над кольцами векторов. В первом разделе также представлены статьи. Посвященные генерации и исследованию псевдослучайных последовательностей битов и функций хеширования.

Во втором разделе представлены статьи по вопросам анализа и синтеза асимметрических криптопреобразований. Не умаляя статьи раздела, прежде всего, отметим статью авторов Качко Е.Г., Балагура Д.С. и Горбенко Ю.И., посвященную обоснование и исследованию практической реализации улучшенного алгоритма цифровой подписи NTRUSign. Полученные авторами результаты имеют важное практическое значение, так как позволяют повысить скорость преобразований с одновременным обеспечением экспоненциальной сложности реализации атаки «полное раскрытие». Заслуживают внимания, скорее теоретического, результаты сравнительного анализа алгоритмов криптографических преобразований основных алгоритмов преобразований на идентификаторах, в которых также применяются алгебраические решетки, авторы профессор Горбенко И.Д. и аспирант Макутонина Л.В. Определенный интерес имеет, скорее теоретический обобщающая статья, Д.А. Паршиной, И.А. Митяевой и И.Д. Горбенко, в которой представлены результаты анализа возможностей применения групп КОС в криптографии. Также во втором разделе публикуются статьи, написанные под руководством профессора Бессалова А.В., тематика которых связанна с приложениями эллиптических кривых. Практическую ценность имеют и остальные статьи второго раздела.

В третьем разделе представлены результаты исследований, связанные с обработкой данных большого объема в условиях неопределенности и практическими исследованиями различных источников биометрической информации.

В четвертом разделе представлены ряд статей, посвященных, различным аспектам защиты информации, в том числе формальные основы методов блокировки аппаратных закладных устройств профессора Горбачева В.А, метод оперативного контроля данных в классе вычетов профессора Краснобаева В.А, рассматривается задача синтеза дискретных сигналов с заданными корреляционными, структурными и ансамблевыми свойствами профессоров Замулы А.А. и Горбенко И.Д, результаты исследований свойств деятельного аспекта защиты информации как системной категории, полученные под руководством профессора Потия А.В., и другие статьи.

С уважением и наилучшими пожеланиями, с благодарностью авторам за подготовленные статьи и будущим читателям. Мы надеемся, что опубликованные в этом журнале статьи послужат решению вопросов развития национальной криптологии.

Ректор ХНУРЭ,
член-корреспондент НАНУ,
профессор

Бондаренко М.Ф.

Заведующий
кафедрой БИТ ХНУРЕ,
профессор

Горбенко И.Д.

СИММЕТРИЧНЫЕ КРИПТОГРАФИЧЕСКИЕ ПРЕОБРАЗОВАНИЯ. СИНТЕЗ И АНАЛИЗ

УДК 621.391:519.2

СУБ'ЕКСПОНЕНЦІЙНІ АЛГОРИТМИ РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ БУЛЕВИХ РІВНЯНЬ ЗІ СПОТВОРЕНИМИ ПРАВИМИ ЧАСТИНАМИ

А.М. ОЛЕКСІЙЧУК

Описано загальну схему побудови відомих суб'експоненційних алгоритмів розв'язання систем лінійних булевих рівнянь зі спотвореними правими частинами. Виділено і проаналізовано найважливіші допоміжні задачі та процедури, що використовуються у зазначених алгоритмах, отримано неасимптотичні оцінки їх надійності. Викладені результати можуть бути використані при розв'язанні ряду задач криптоаналізу і теорії вивідування.

Ключові слова: система лінійних рівнянь зі спотвореними правими частинами, задача про адитивне представлення, суб'експоненційний алгоритм, кореляційний криптоаналіз.

ВСТУП

Системи лінійних булевих рівнянь зі спотвореними правими частинами є традиційним об'єктом досліджень в теорії кодування та криптології [1, 2]. Розв'язання таких систем загального вигляду рівносильно декодуванню довільних двійкових лінійних кодів. Остання задача є NP-повною [3] і для неї не відомо (та, ймовірно, не існує) поліноміальних алгоритмів.

Дана стаття присвячена аналізу алгоритмів, які дозволяють розв'язувати системи лінійних булевих рівнянь зі спотвореними правими частинами від n невідомих за суб'експоненційний час, тобто за $2^{o(n)}$ операцій над n-вимірними двійковими векторами. Перший такий алгоритм запропоновано в 1988 р. І. М. Коваленком [4], який показав, що у випадку, коли матриця коефіцієнтів системи, яка складається з $2^{O\left(\frac{n}{\log n}\right)}$ лінійних рівнянь зі спотвореними правими частинами, є випадковою та задовольняє певній загальній умові, зазначену систему можна розв'язати з як завгодно високою при $n \to \infty$ надійністю, використовуючи в середньому $2^{O\left(\frac{n}{\log n}\right)}$ операцій.

Алгоритм Коваленка [4] було фактично перевідкрито тринадцять років потому А. Блюмом, А. Калаі та Х. Вассерманом [5], які вивчали задачу розв'язання систем лінійних булевих рівнянь зі спотвореними правими частинами та випадковими рівноймовірними матрицями коефіцієнтів у зв'язку з однією проблемою теорії вивідування (learning theory). Алгоритм Блюма-Калаі-Вассермана (BKW) [5] є більш простим у порівнянні з алгоритмом Коваленка. Він швидко набув широкої відомості та знайшов чимало застосувань як у теорії вивідування, так і за її межами. Зауважимо, що обидва алгоритми є застосовними лише до таких систем, які складаються з великої кількості (а саме, з $2^{O\left(\frac{n}{\log n}\right)}$) лінійних рівнянь.

У 2005 р. В. Любашевський [6] показав, що задача розв'язання системи з $n^{1+\alpha}$ лінійних рівнянь від n невідомих зі спотвореними правими частинами та рівноймовірною матрицею коефіцієнтів зводиться до аналогічної задачі для системи з $2^{O\left(\frac{n}{\log n}\right)}$ лінійних рівнянь і запропонував алгоритм розв'язання зазначених систем (які складаються з $n^{1+\alpha}$ лінійних рівнянь, $\alpha > 0$) зі складністю $2^{O\left(\frac{n}{\log\log n}\right)}$. Відзначені результати робіт [5, 6] узагальнені С. Коппарті та С. Сарафом [7] на випадок систем лінійних булевих рівнянь з "агностичними" спотвореннями, тобто таких систем рівнянь, праві частини яких можуть формуватися довільним чином.

Ідеї та результати, викладені у відзначених (та деяких інших) публікаціях, дозволяють встановити загальну схему, за якою будуються відомі суб'експоненційні алгоритми розв'язання систем лінійних рівнянь зі спотвореними правими частинами, виділити важливі задачі та проаналізувати окремі процедури, що використовуються у зазначених алгоритмах, отримати неасимптотичні оцінки їх надійності. Викладення відзначених результатів є метою даної статті.

Основні результати статті сформульовані у вигляді двох "теорем зведення". Перша теорема базується на ідеях роботи [5] та дає відповідь на запитання, як саме (з якими надійністю і трудомісткістю) можна розв'язувати системи лінійних рівнянь зі спотвореними правими частинами на основі заданого алгоритму розв'язання так званої задачі про адитивне представлення. Друга теорема базується на ідеях робіт [6, 7] і показує, як розв'язувати системи з невеликої кількості лінійних рівнянь зі спотвореними правими частинами на основі заданого алгоритму розв'язання таких систем, що складаються з великої кількості рівнянь. Проаналізовано також найважливіші властивості процедури самокорекції [7], яка може

бути корисною не тільки для розв'язання систем лінійних рівнянь зі спотвореннями. Сформульовано ряд задач подальших досліджень.

1. ЗВЕДЕННЯ ЗАДАЧІ РОЗВ'ЯЗАННЯ СИСТЕМИ ЛІНІЙНИХ РІВНЯНЬ ЗІ СПОТВОРЕНИМИ ПРАВИМИ ЧАСТИНАМИ ДО ЗАДАЧІ ПРО АДИТИВНЕ ПРЕДСТАВЛЕННЯ

Задача про *адитивне r-представлення* (або *r-суму*, *r-sum problem*) полягає в наступному [8]. Задано список, тобто впорядкований набір L, що складається з l векторів $z_1,\dots,z_l \in V_n = \{0,1\}^n$. Потрібно для будь-якого вектора $z \in V_n$ знайти r (не обов'язково різних) номерів $\nu_1,\dots,\nu_r \in \overline{1,l}$ таких, що $z_{\nu_1} \oplus \dots \oplus z_{\nu_r} = z$. Припускається, що будь-який алгоритм **A** розв'язання цієї задачі або завершується успішно, тобто знаходить шуканий набір $\nu_1,\dots,\nu_r$, або видає негативну відповідь, тобто не знаходить зазначеного набору (хоча такий може існувати).

Нехай вектори $z_1,\dots,z_l$ у списку L генеруються незалежно один від одного у відповідності з певним розподілом ймовірностей P на множині V_n. Мінімальна за всіма векторами $z \in V_n$ ймовірність успішного завершення алгоритму **A** при вхідних даних (L,z) називається *P-надійністю алгоритму* **A** та позначається $\pi_{\mathbf{A},P} = \pi_{\mathbf{A},P}(n,r,l)$. Якщо P є рівномірним розподілом на множині V_n, то P-надійність алгоритму **A** називається його *надійністю* та позначається $\pi_{\mathbf{A}}$. *Трудомісткість алгоритму* **A** визначається як найбільше число операцій над n-вимірними двійковими векторами, що виконуються при його застосуванні до будь-яких вхідних даних (L,z), та позначається $T_{\mathbf{A}} = T_{\mathbf{A}}(n,r,l)$.

Відзначимо окремий випадок задачі про адитивне r-представлення, в якому задається r вхідних списків $L_1,\dots,L_r$ довжини $l_1,\dots,l_r$ відповідно, що складаються з незалежних в сукупності випадкових та рівноймовірних двійкових векторів довжини n. Потрібно знайти набір $z_1 \in L_1,\dots,z_r \in L_r$ такий, що $z_1 \oplus \dots \oplus z_r = 0$ [9, 10]. Зрозуміло, що при $l_1 + \dots + l_r = l$ будь-який алгоритм **A** розв'язання останньої задачі дозволяє розв'язувати першу: достатньо розбити вхідний список L на r частин $L_1, L_2, \dots, L_r$ та застосувати алгоритм **A** до списків $L_1 \oplus z, L_2, \dots, L_r$.

Задача про адитивне представлення добре відома в теорії кодування, теорії обчислювальних алгоритмів та криптоаналізі [8–10]. Зокрема, на ній, тією чи іншою мірою, базуються відомі суб'експоненційні алгоритми розв'язання систем лінійних булевих рівнянь зі спотвореними правими частинами. Нижче доводиться загальна теорема, яка дозволяє будувати алгоритми розв'язання зазначених систем рівнянь на основі будь-яких алгоритмів розв'язання задачі про адитивне представлення.

Розглянемо систему рівнянь

$$Ax = b, \tag{1}$$

де A – булева матриця розміру $m \times n$, b – вектор довжини m з координатами

$$b_i = A_i a \oplus \xi_i,\ i \in \overline{1,m}, \tag{2}$$

де $A_1,\dots,A_m$ – рядки матриці A, $a = (a_1,\dots,a_n)^T$ – невідомий двійковий вектор (істинний розв'язок системи рівнянь (1)), $\xi_1,\dots,\xi_m$ – незалежні випадкові величини, розподілені за законами

$$\mathrm{P}\{\xi_i = 0\} = 1 - \mathrm{P}\{\xi_i = 1\} = \frac{1}{2}(1 + \theta_i),\ i \in \overline{1,m}, \tag{3}$$

де $\theta_i \ge \theta > 0$ для кожного $i \in \overline{1,m}$.

Позначимо e_i вектор довжини n, i-а координата якого дорівнює 1, а решта – 0, $i \in \overline{1,n}$. Сформулюємо допоміжне твердження, яке неодноразово використовується далі.

Лема 1 (*нерівність Гефдінга*) [11]. Нехай $\zeta_1,\dots,\zeta_t$ – незалежні випадкові величини такі, що $\alpha_j \le \zeta_j \le \beta_j$, $\alpha_j, \beta_j \in \mathbf{R}$, $j \in \overline{1,t}$. Тоді для будь-якого $x > 0$

$$\mathbf{P}\{\sum_{l=1}^{t}\zeta_l - \sum_{l=1}^{t}\mathbf{E}\zeta_l \ge tx\} \le \exp\left\{-\frac{2t^2x^2}{\sum_{l=1}^{t}(\beta_l - \alpha_l)^2}\right\}.$$

Наступна теорема базується на узагальненні міркувань, що використовуються при доведенні теореми 3 у статті [5].

Теорема 1. Припустимо, що матриця A системи рівнянь (1) складається з $m = nlt$ рядків, які є незалежними випадковими векторами, що мають однаковий розподіл ймовірностей P на множині V_n і не залежать від випадкових величин $\xi_1,\dots,\xi_m$. Тоді для будь-якого алгоритму **A** розв'язання задачі про адитивне представлення з параметрами n, r, l (див. вище) існує алгоритм **B**, який знаходить істинний розв'язок системи рівнянь (1) з імовірністю

$$\pi_{\mathbf{B}}(P,\theta) \ge (\pi_{\mathbf{A},P})^{tn}(1 - \exp\{-1/2 \cdot \theta^{2r}t\})^n, \tag{4}$$

використовуючи

$$T_{\mathbf{B}} = O(nt(T_{\mathbf{A}} + r)) \tag{5}$$

операцій над n-вимірними двійковими векторами.

Доведення. Алгоритм **B**, що пропонується, має такий вигляд.

1. Розіб'ємо систему рядків матриці A на tn списків $L_{i,j}$ довжини l кожний та застосуємо алгоритм **A** до вхідних даних $(L_{i,j}, e_i)$ для всіх $i \in \overline{1,n}$, $j \in \overline{1,t}$. Якщо хоча б в одному випадку алгоритм **A** завершується неуспішно, то алгоритм **B** припиняє роботу. Інакше для кожного $i \in \overline{1,n}$ отримаємо рівності вигляду

$$e_i = A_{\nu_1(i,j)} \oplus \dots \oplus A_{\nu_r(i,j)},\ j \in \overline{1,t}, \tag{6}$$

де $A_{\nu_1(i,j)},\dots,A_{\nu_r(i,j)} \in L_{i,j}$.

2. Для будь-яких $i \in \overline{1,n}$, $j \in \overline{1,t}$ обчислимо значення $b(i,j) = b_{\nu_1(i,j)} \oplus \ldots \oplus b_{\nu_r(i,j)}$ та відновимо i-у координату вектора a за мажоритарним правилом:

$$a_i = 1 \Leftrightarrow \sum_{j=1}^{t} b(i,j) \geq \frac{t}{2}.$$

Безпосередньо з наведеного опису випливає, що трудомісткість алгоритму **B** визначається за формулою (5).

Доведемо формулу (4). Нехай A – випадкова $m \times n$-матриця з незалежними рядками, що розподілені на множині V_n за законом P. Тоді усі списки $L_{i,j}$, які формуються на кроці 1 алгоритму, складаються з незалежних в сукупності випадкових векторів, що мають такий самий закон розподілу.

Позначимо $\Im$ множину значень випадкової матриці A, для кожного з яких алгоритм **A** завершується успішно при всіх його застосуваннях на кроці 1. Далі, для кожного $A^{(0)} \in \Im$ позначимо $\Re(A^{(0)})$ подію, яка полягає в тому, що алгоритм **B** вірно відновлює вектор a на кроці 2 за умови, що $A = A_0$. В силу незалежності матриці A та випадкових величин $\xi_1, \ldots, \xi_m$ ймовірність вірного відновлення вектора a дорівнює

$$\pi_{\mathbf{B}}(P,\theta) = \sum_{A^{(0)} \in \Im} \mathbf{P}\{A = A^{(0)}\}\, \mathbf{P}\{\Re(A^{(0)})\}. \qquad (7)$$

Зафіксуємо матрицю $A^{(0)} \in \Im$ та оцінимо ймовірність події $\Re(A^{(0)})$. Позначимо $L_{i,j}^{(0)}$, $i \in \overline{1,n}$, $j \in \overline{1,t}$, списки, які формується на кроці 1 алгоритму **B** за вхідною матрицею $A^{(0)}$. Помітимо, що на підставі формул (2) та (6) справедливі наступні рівності:

$$a_i = e_i a = b(i,j) \oplus (\xi_{\nu_1(i,j)} \oplus \cdots \oplus \xi_{\nu_r(i,j)}),$$
$$i \in \overline{1,n},\ j \in \overline{1,t}.$$

При цьому числа $\nu_1(i,j), \ldots, \nu_r(i,j)$ є (не обов'язково різними) номерами рядків, що належать списку $L_{i,j}^{(0)}$. Отже, для будь-яких $(i,j) \neq (i',j')$ справедливе співвідношення

$$\{\nu_1(i,j), \ldots, \nu_r(i,j)\} \cap \{\nu_1(i',j'), \ldots, \nu_r(i',j')\} = \varnothing,$$

з якого випливає, що випадкові величини $\xi_{\nu_1(i,j)} \oplus \cdots \oplus \xi_{\nu_r(i,j)}$, $i \in \overline{1,n}$, $j \in \overline{1,t}$, є незалежними в сукупності. Нарешті, на підставі формули (3) та умови $\theta_s \geq \theta > 0$, $s \in \overline{1,m}$, виконується нерівність

$$\mathbf{P}\{\xi_{\nu_1(i,j)} \oplus \cdots \oplus \xi_{\nu_r(i,j)} = 0\} \geq \frac{1}{2}\left(1 + \theta^r\right),$$
$$i \in \overline{1,n},\ j \in \overline{1,t}.$$

Таким чином, для оцінки ймовірності події $\Re(A^{(0)})$ можна скористатися нерівністю Гефдінга (див. лему 1). Згідно з цією нерівністю, ймовірність помилки при відновленні кожного окремого значення a_i, $i \in \overline{1,n}$, на другому кроці алгоритму **B** не перевищує $\exp\{-1/2 \cdot \theta^{2r} t\}$. Отже,

$$\mathbf{P}\{\Re(A^{(0)})\} \geq (1 - \exp\{-1/2 \cdot \theta^{2r} t\})^n.$$

Підставляючи зазначену оцінку в формулу (7), з урахуванням нерівності $\mathbf{P}\{A \in \Im\} \geq (\pi_{\mathbf{A},P})^n$ отримаємо формулу (4). Теорему доведено.

Як видно з доведення, теорема залишається справедливою і в тому випадку, коли алгоритм **A** дозволяє знаходити з ймовірністю не менше за $\pi_{\mathbf{A},P}$ адитивне r-представлення кожного з векторів $e_1, \ldots, e_n$ (але не обов'язково довільного вектора $z \in V_n$). Наведемо конкретний приклад такого алгоритму, що запропоновано в [5].

Лема 2 (*алгоритм BKW*). Нехай u, v, λ – натуральні числа і

$$n \leq uv,\ r = 2^{u-1},\ l = (u + \lambda - 1)2^v. \qquad (8)$$

Тоді існує алгоритм $\mathbf{A}_0$, який знаходить адитивне r-представлення кожного з векторів $e_1, \ldots, e_n$ у випадковому рівноймовірному списку L довжини l з надійністю $\pi_{\mathbf{A}_0} \geq 1 - e^{-\lambda}$, використовуючи $T_{\mathbf{A}_0} = O(u(u+\lambda)2^v)$ операцій над n-вимірними двійковими векторами.

Доведення. Опишемо алгоритм знаходження r-представлення вектора e_1. Адитивні представлення векторів $e_2, \ldots, e_n$ можна побудувати, застосовуючи зазначений алгоритм до списків, які отримуються шляхом циклічного зсуву всіх векторів зі списку L на $1, \ldots, n-1$ позицій відповідно.

Не обмежуючи загальності, вважатимемо, що $n = uv$ (у протилежному випадку допишемо до кожного вектора довжини n необхідну кількість нулів). Отже, будь-який вектор $z \in V_n$ можна розглядати як послідовність u двійкових слів довжини v біт кожне та записувати у вигляді $z = (z^{(1)}, \ldots, z^{(u)})$, де $z^{(i)} \in V_v$, $i \in \overline{1,u}$.

Алгоритм $\mathbf{A}_0$ знаходження адитивного r-представлення вектора e_1 у випадковому рівноймовірному списку $L^{(i)}$ має такий вигляд.

Розіб'ємо вхідний список на блоки, відносячи до одного і того ж блоку L_c ($c \in V_v$) усі вектори $z \in L$ такі, що $z^{(u)} = c$. Зауважимо, що зазначене розбиття можна отримати, використовуючи $O(l)$ операцій, де l – довжина списку L.

Далі для кожного непорожнього блоку L_c виконаємо наступну процедуру: виберемо з блоку L_c випадково та рівноймовірно один вектор, додамо його до кожного іншого вектора з цього блоку та вилучимо зі списку L. В результаті отримаємо новий список $L^{(1)}$, що складається не менше ніж з $l_1 = l - 2^v$ векторів, які задовольняють наступним умовам:

(а) для будь-якого $z \in L^{(1)}$ виконується рівність $z^{(u)} = 0$;

(б) кожен вектор $z \in L^{(1)}$ є сумою точно двох векторів зі списку L;

(в) підвектори, що складаються з перших $u-1$ слів усіх векторів зі списку $L^{(1)}$, є незалежними в сукупності випадковими рівноймовірними векторами довжини $(u-1)v$.

Справедливість тверджень (а) – (в) випливає безпосередньо з наведеного опису процедури формування списку $L^{(1)}$ за списком L і умови випадковості та рівноймовірності останнього.

Далі застосуємо аналогічну процедуру до списку $L^{(1)}$ та отримаємо список $L^{(2)}$, що складається не менше ніж з $l_2 = l - 2^v - 2^v$ векторів, які задовольняють умовам, аналогічним (а) – (в). Продовжуючи зазначений процес, отримаємо послідовність списків $L^{(1)}$, ..., $L^{(u-1)}$ таких, що для кожного $i \in \overline{1,u-1}$ список $L^{(i)}$ складається не менше ніж з $l - i2^v$ векторів z, кожен з яких задовольняє умові $z^{(u-i+1)} = ... = z^{(u)} = 0$ та є сумою точно 2^i векторів зі списку L. При цьому підвектори, що складаються з перших $u-i$ слів усіх векторів зі списку $L^{(i)}$, є незалежними в сукупності випадковими рівноймовірними двійковими векторами довжини $(u-i)v$.

На останньому кроці, при $i = u-1$, отримаємо список $\tilde{L}$, який складається не менше ніж з $\lambda 2^v = l - (u-1)2^v$ незалежних випадкових та рівноймовірних векторів $z^{(1)} \in V_v$ таких, що вектори $(z^{(1)}, 0, ..., 0)$ утворюють список $L^{(u-1)}$. Оскільки ймовірність появи будь-якого фіксованого вектора довжини v у списку $\tilde{L}$ є не менше за $1 - \left(\frac{2^v - 1}{2^v}\right)^{\lambda 2^v} \geq 1 - e^{-\lambda}$, то вектор e_1 зустрінеться у списку $L^{(u-1)}$ з такою самою ймовірністю. При цьому, оскільки кожен вектор з останнього списку є сумою точно $r = 2^{u-1}$ векторів, що належать списку L, то шукане адитивне r-представлення вектора e_1 можна отримати з імовірністю $\pi_{\mathbf{A}_0} \geq 1 - e^{-\lambda}$.

Нарешті, як випливає з наведеного опису алгоритму $\mathbf{A}_0$, його трудомісткість складає $T_{\mathbf{A}_0} = O(ul) = O(u(u+\lambda)2^v)$ операцій. Лему доведено.

В [5] рекомендується вибирати значення параметрів u, v наступним чином:

$$u = \left\lceil \frac{\log n}{2} \right\rceil, \; v = \left\lceil \frac{2n}{\log n} \right\rceil.$$

Нехай для простоти $n = 2^{2^s}$, де $s \in \mathbf{N}$. Тоді

$$u = \frac{\log n}{2}, \; v = \frac{2n}{\log n} \qquad (9)$$

і, згідно зі співвідношеннями (8),

$$2r = \sqrt{n}, \; l = \left(\frac{\log n}{2} + \lambda - 1\right) 2^{\frac{2n}{\log n}}. \qquad (10)$$

Розглянемо систему рівнянь (1), що задовольняє умовам (2) та (3). Як і вище, припустимо, що $\theta_i \geq \theta > 0$ для кожного $i \in \overline{1,m}$ та покладемо

$$t = \left\lceil 2C\theta^{-\sqrt{n}} \ln n \right\rceil, \; \lambda = \left\lceil \ln(tn\delta^{-1}) \right\rceil, \qquad (11)$$

вважаючи, що величини $C > 1$, $\delta \in (0,1)$ і θ є константами (тобто не залежать від n). Позначимо $\mathbf{B}_0$ алгоритм розв'язання системи рівнянь (1) з випадковою рівноймовірною матрицею коефіцієнтів, який будується за алгоритмом $\mathbf{A}_0$ у відповідності з доведенням теореми 1. На підставі цієї теореми та рівностей (9) – (11) справедливі такі співвідношення:

$$m = nlt = O\left(n^{3/2} \log n \log(\theta^{-1}) \, \theta^{-\sqrt{n}} 2^{\frac{2n}{\log n}} \right) = 2^{O\left(\frac{n}{\log n}\right)}, \qquad (12)$$

$$\pi_{\mathbf{B}_0} \geq 1 - tne^{-\lambda} - n\exp\{-1/2 \cdot \theta^{2r} t\} \geq 1 - \delta - n^{1-C}, \qquad (13)$$

$$T_{\mathbf{B}_0} = O\left(n^{3/2} \log^2 n \log(\theta^{-1}) \, \theta^{-\sqrt{n}} 2^{\frac{2n}{\log n}} \right) = 2^{O\left(\frac{n}{\log n}\right)}. \qquad (14)$$

Отже, доведено наступне твердження.

Наслідок 1. За виконанням співвідношень (9) – (12) існує алгоритм, який знаходить істинний розв'язок системи рівнянь (1) зі спотвореною правою та випадковою рівноймовірною лівою частинами з надійністю (13) і трудомісткістю (14).

Відзначимо, що наслідок 1 випливає з основного результату статті [4] і доводиться в [5] (за винятком оцінки надійності алгоритму).

Спираючись на теорему 1, можна побудувати алгоритми розв'язання систем лінійних рівнянь зі спотвореними правими частинами, що базуються на узагальненому алгоритмі Вагнера [9, 10]. Цей алгоритм призначено для знаходження адитивного r-представлення нульового вектора за r незалежними випадковими рівноймовірними списками та є за сутністю близьким до алгоритму BKW. Для зменшення трудомісткості знаходження r-представлення в [10] пропонується розбивати n-вимірні вектори на слова різної довжини (зауважимо, що аналогічна ідея використовується в [12] для побудови оптимальних у певному класі модифікацій алгоритму Коновальцева).

Розв'язання задачі про адитивне представлення є найбільш трудомістким етапом знаходження істинного розв'язку системи рівнянь (1). Отже, будь-який прогрес у вирішенні цієї задачі призведе до зменшення складності алгоритмів розв'язання систем лінійних рівнянь зі спотвореними правими частинами. Найкращі з відомих сьогодні алгоритмів мають ті ж самі асимптотичні характеристики, що й алгоритм, зазначений у наслідку 1: трудомісткість розв'язання системи з $2^{O\left(\frac{n}{\log n}\right)}$ лінійних рівнянь від n невідомих зі спотвореними правими частинами та випадковою рівноймовірною матрицею коефіцієнтів складає $2^{O\left(\frac{n}{\log n}\right)}$ операцій.

2. ПРОЦЕДУРА САМОКОРЕКЦІЇ

У випадку, коли система (1) містить невелику кількість рівнянь або розподіл її матриці коефіцієнтів помітно відрізняється від рівномірного, можна застосовувати так звану процедуру самокорекції, яка має за мету звести знаходження істинного розв'язку цієї системи рівнянь до

розв'язання задачі, що розглянута в попередньому пункті.

Термін "самокорекція" запропоновано в [7] і позначає певну процедуру, що дозволяє отримувати з невеликої кількості незалежних випадкових двійкових векторів, які мають "не надто поганий" розподіл, довільну кількість незалежних векторів, що розподілені "майже" рівномірно.

Сформулюємо точне означення. Нехай G – матриця розміру $T\times n$ з рядками $G_1,\dots,G_T\in V_n$, χ – вектор довжини T з координатами $\chi_1,\dots,\chi_T\in\{0,1\}$. *Процедура самокорекції з параметрами* (m,k), де $m,k\in\mathbf{N}$, полягає у застосуванні до вхідних даних (G,χ) наступного алгоритму.

Для кожного $i\in\overline{1,m}$:

1) згенерувати незалежні в сукупності випадкові величини $\mu_{1,i},\dots,\mu_{k,i}$ з рівномірним розподілом на множині $\overline{1,T}$;

2) обчислити

$$X_i=G_{\mu_{1,i}}\oplus\dots\oplus G_{\mu_{k,i}},\ \xi_i=\chi_{\mu_{1,i}}\oplus\dots\oplus\chi_{\mu_{k,i}}.$$

Результатом виконання алгоритму є список, що складається з m випадкових векторів (X_i,ξ_i), $i\in\overline{1,m}$. Зрозуміло, що зазначені вектори є незалежними в сукупності та однаково розподіленими на множині V_{n+1}, і основне питання полягає в тому, за яких умов вектор X_i має "майже" рівномірний розподіл на V_n, а випадкова величина ξ_i "майже" не залежить від вектора X_i. Більш точно, позначимо $P_{X,\xi}$ сумісний розподіл ймовірностей випадкових елементів $X=G_{\mu_1}\oplus\dots\oplus G_{\mu_k}$ та $\xi=\chi_{\mu_1}\oplus\dots\oplus\chi_{\mu_k}$ (опускаючи для простоти позначень індекс i). Розглянемо також розподіл ймовірностей $U_\xi(x,u)=2^{-n}\mathbf{P}\{\xi=u\}$, $(x,u)\in V_{n+1}$. Потрібно оцінити відстань по варіації між зазначеними розподілами на множині V_{n+1}.

Нагадаємо, що відстань по варіації між розподілами ймовірностей P та Q на довільній скінченній множині Ω визначається за формулою

$$d(P,Q)=\frac{1}{2}\sum_{\omega\in\Omega}|P(\omega)-Q(\omega)|.$$

Наступне добре відоме твердження, доведення якого неважко отримати безпосередньо з означення відстані по варіації, містить основні властивості цього параметра, що використовуються далі.

Лема 3 (*властивості відстані по варіації*).

1. Для будь-яких розподілів ймовірностей P, Q на скінченній множині Ω справедлива рівність $d(P,Q)=\max\limits_{B\subseteq\Omega}\{|P(B)-Q(B)|\}$.

2. Нехай $\Omega=\Omega_1\times\dots\times\Omega_m$, P_i,Q_i – розподіли ймовірностей на множині Ω_i, $i\in\overline{1,m}$. Позначимо $P=P_1\times\dots\times P_m$, $Q=Q_1\times\dots\times Q_m$. Тоді

$$d(P,Q)\le\sum_{i=1}^m d(P_i,Q_i).$$

Доведемо зараз лему, яка встановлює верхню оцінку відстані по варіації між розподілами ймовірностей $P_{X,\xi}$ та U_ξ.

Лема 4. Нехай G – $T\times n$-матриця з рядками $G_1,\dots,G_T\in V_n$, χ – T-вимірний вектор з координатами $\chi_1,\dots,\chi_T\in\{0,1\}$, $X=G_{\mu_1}\oplus\dots\oplus G_{\mu_k}$, $\xi=\chi_{\mu_1}\oplus\dots\oplus\chi_{\mu_k}$, де $\mu_1,\dots,\mu_k$ – незалежні випадкові величини з рівномірним розподілом на множині $\overline{1,T}$. Тоді справедлива нерівність

$$d(P_{X,\xi},U_\xi)=\frac{1}{2}\sum_{\substack{x\in V_n,\\ u\in\{0,1\}}}|\mathbf{P}\{X=x,\xi=u\}-2^{-n}\mathbf{P}\{\xi=u\}|\le$$

$$\le 2^n(\Delta_{G,\chi})^k, \qquad (15)$$

де

$$\Delta_{G,\chi}=T^{-1}\max_{\substack{y\in V_n\setminus\{0\},\\ u\in\{0,1\}}}\left|\sum_{j=1}^T(-1)^{G_jy\oplus u\chi_j}\right|. \qquad (16)$$

Доведення. Позначимо

$$\varphi_k(y,w)=\left(T^{-1}\sum_{j=1}^T(-1)^{G_jy\oplus w\chi_j}\right)^k,\ (y,w)\in V_{n+1}. \qquad (17)$$

За допомогою безпосередньої перевірки неважко переконатися в тому, що для будь-якого $(x,u)\in V_{n+1}$ виконуються рівності

$$\mathbf{P}\{X=x,\xi=u\}=2^{-(n+1)}\sum_{(y,w)\in V_{n+1}}(-1)^{xy\oplus uw}\varphi_k(y,w),$$

$$\mathbf{P}\{\xi=u\}=\frac{1}{2}\left(1+(-1)^u\left(T^{-1}\sum_{j=1}^T(-1)^{\chi_j}\right)^k\right)=$$

$$=\frac{1}{2}\left(1+(-1)^u\varphi_k(0,1)\right). \qquad (18)$$

Звідси випливає, що

$$\mathbf{P}\{X=x,\xi=u\}-2^{-n}\mathbf{P}\{\xi=u\}=2^{-(n+1)}\times$$

$$\times\left(\sum_{y\in V_n\setminus\{0\}}(-1)^{xy}\varphi_k(y,0)+(-1)^u\sum_{y\in V_n\setminus\{0\}}(-1)^{xy}\varphi_k(y,1)\right)$$

і, отже, на підставі формул (16) та (17)

$$|\mathbf{P}\{X=x,\xi=u\}-2^{-n}\mathbf{P}\{\xi=u\}|\le 2^{-(n+1)}\times$$

$$\times\left(2^n\max_{y\in V_n\setminus\{0\}}|\varphi_k(y,0)|+2^n\max_{y\in V_n\setminus\{0\}}|\varphi_k(y,1)|\right)\le$$

$$\le(\Delta_{G,\chi})^k.$$

З отриманої нерівності випливає формула (15). Лему доведено.

Отже, близкість сумісного розподілу випадкових елементів X та ξ до розподілу ймовірностей U_ξ на множині V_{n+1} визначається параметром (16). Відзначимо теоретико-кодовий сенс цього параметра.

Позначимо $C(G,\chi)$ лінійний код з твірною матрицею $\begin{pmatrix}G^T\\ \chi\end{pmatrix}$, тобто код довжини T, що складається зі слів вигляду $Gy\oplus u\chi$, де $(y,u)\in V_{n+1}$. Неважко бачити, що величина $T^{-1}\sum\limits_{j=1}^T(-1)^{G_jy\oplus u\chi_j}$ дорівнює $1-2wt(Gy\oplus u\chi)$, де є $wt(Gy\oplus u\chi)$ є

відносною вагою слова $Gy \oplus u\chi$ коду $C(G,\chi)$. Отже, параметр $\Delta_{G,\chi}$ співпадає з найменшим числом $\Delta \in [0,1]$, для якого відносна вага кожного слова $Gy \oplus u\chi$ коду $C(G,\chi)$, де $y \neq 0$, знаходиться в межах від $\frac{1}{2}(1-\Delta)$ до $\frac{1}{2}(1+\Delta)$.

Назвемо код $C(G,\chi)$ γ *-збалансованим*, якщо виконується умова $\Delta_{G,\chi} \le T^{-\gamma}$, де $\gamma > 0$.

Безпосередньо з леми 4 випливає такий результат.

Наслідок 2. Нехай $C(G,\chi) \in$ γ -збалансованим кодом,

$$k = \left\lceil \frac{(1+c)n}{\gamma \log T} \right\rceil,\ c > 0. \qquad (19)$$

Тоді в умовах леми 4 справедлива нерівність

$$d(P_{X,\xi}, U_\xi) \le 2^{-cn}. \qquad (20)$$

Отже, для γ -збалансованих кодів відстань по варіації між розподілами ймовірностей $P_{X,\xi}$ та U_ξ на множині V_{n+1} експоненційно швидко прямує до нуля при $n \to \infty$, якщо k визначається за формулою (19), де $c = const$.

Як показує наступна лема, властивістю γ -збалансованості з високою ймовірністю володіють певні випадкові коди.

Лема 5. Нехай рядки $T \times n$ -матриці G є незалежними випадковими рівноймовірними векторами на множині V_n, а координати вектора χ – незалежними випадковими величинами, що не залежать від матриці G. Нехай, далі, $T = n^{1+\alpha}$, де $\alpha > 0$,

$$\gamma = \frac{\alpha(1-c)}{2(1+\alpha)},\ 0 < c < 1. \qquad (21)$$

Тоді випадковий код $C(G,\chi)$ є γ -збалансованим з імовірністю біля $1 - 2^{1-n}$.

Доведення. З умови леми випливає, що для будь-яких $y \in V_n \setminus \{0\}$, $u \in \{0,1\}$ випадкові величини $(-1)^{G_j y \oplus u\chi_j}$, $j \in \overline{1,T}$, є незалежними та рівноймовірними. Отже, на підставі леми 1

$$\mathbf{P}\{\Delta_{G,\chi} > T^{-\gamma}\} =$$

$$= \mathbf{P}\left\{ \max_{y \in V_n \setminus \{0\}} \left| T^{-1} \sum_{j=1}^{T} (-1)^{G_j y \oplus u\chi_j} \right| > T^{-\gamma} \right\} <$$

$$< 2^n \cdot 2\exp\{-2T(T^{-\gamma})^2\} < 2^{n+1-2T^{1-2\gamma}}.$$

Далі, згідно з рівністю $T = n^{1+\alpha}$ та формулою (21),

$$n + 1 - 2T^{1-2\gamma} = n + 1 - 2n^{(1+\alpha)(1-2\gamma)} < n + 1 - 2n = 1 - n,$$

звідки випливає, що $\mathrm{P}\{\Delta_{G,\chi} > T^{-\gamma}\} < 2^{1-n}$. Лему доведено.

3. ЗАСТОСУВАННЯ САМОКОРЕКЦІЇ ДО РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ РІВНЯНЬ ЗІ СПОТВОРЕНИМИ ПРАВИМИ ЧАСТИНАМИ

Даний пункт присвячено доведенню теореми Любашевського [6] про можливість розв'язання за суб'експоненційний час системи лінійних рівнянь зі спотвореними правими частинами та випадковою рівноймовірною матрицею коефіцієнтів розміру $T \times n$, де $T = n^{1+\alpha}$, $\alpha > 0$. Доведення, що наводиться нижче, використовує основну ідею роботи [6], але проводиться іншим, більш простим методом. Наводиться також вираз оцінки надійності алгоритму розв'язання заданої системи рівнянь, який відсутній у [6].

Розглянемо систему рівнянь

$$Gx = h, \qquad (22)$$

де G – випадкова рівноймовірна булева матриця розміру $T \times n$, h – вектор з координатами

$$h_i = G_i a \oplus \chi_i,\ i \in \overline{1,T}, \qquad (23)$$

$G_1, \ldots, G_T$ – рядки матриці G, $a = (a_1, \ldots, a_n)^T \in V_n$ – невідомий істинний розв'язок системи рівнянь (22), $\chi_1, \ldots, \chi_T$ – незалежні випадкові величини, розподілені за законами

$$\mathbf{P}\{\chi_i = 0\} = 1 - \mathbf{P}\{\chi_i = 1\} = \frac{1}{2}(1 + \theta_i),\ i \in \overline{1,T}, \qquad (24)$$

де $\theta_i \ge \theta > 0$ для кожного $i \in \overline{1,T}$.

Ідея побудування суб'експоненційного алгоритму розв'язання системи (22) полягає в наступному. Застосуємо до вхідних даних (G,h) процедуру самокорекції з параметрами (m,k) та подамо отриманий список векторів

$$(A_i = G_{\mu_{1,i}} \oplus \cdots \oplus G_{\mu_{k,i}},$$

$$b_i = A_i a \oplus (\chi_{\mu_{1,i}} \oplus \cdots \oplus \chi_{\mu_{k,i}}))\ ,\ i \in \overline{1,m}, \qquad (25)$$

на вхід довільного алгоритму **B**, який дозволяє розв'язувати системи рівнянь вигляду (1) з певній надійністю $\pi_{\mathbf{B}}(\theta)$ за суб'експоненційний від n час. Згідно з наслідком 2 та лемою 5, вибираючи належним чином параметр k, можна добитися того, щоб випадкові вектори A_i були "майже" рівноймовірними на множині V_n, а випадкові величини $\xi_i = \chi_{\mu_{1,i}} \oplus \cdots \oplus \chi_{\mu_{k,i}}$ "майже" не залежили від A_i, $i \in \overline{1,m}$. Звідси, спираючись на лему 3, неважко вивести, що ймовірність вірного відновлення вектора a з отриманої після самокорекції системи рівнянь "майже" не відрізняється від $\pi_{\mathbf{B}}(\theta^k)$. Нарешті, спираючись на теорему 1, можна вибрати параметр m таким чином, щоби сумарний час відновлення вектора a (за допомогою процедури самокорекції та алгоритму **B**) суб'експоненційно залежав від n.

В [6] пропонується використовувати в ролі **B** алгоритм із [5], а замість самокорекції застосовувати декілька іншу процедуру, яка полягає в додаванні k різних рядків розширеної матриці системи (22), що вибираються за урновою схемою без повернення. Зазначимо, що аналогічна процедура використовується на першому кроці алгоритму Коваленка [4], а також у деяких інших алгоритмах розв'язання систем лінійних рівнянь зі спотвореними правими частинами [13]. Проте обидві ймовірнісні схеми (рівноймовірного вибору рядків з поверненням чи без нього) приводять

до однакових асимптотичних оцінок трудомісткості алгоритмів, а незалежний рівноймовірний вибір рядків, що здійснюється при виконанні самокорекції, дозволяє помітно спростити ймовірнісний аналіз.

Перейдемо до більш точного викладення наведених міркувань. Перш за все, доведемо просту лему, яка встановлює зв'язок між значеннями надійності будь-якого алгоритму розв'язання системи рівнянь (1) при різних розподілах ймовірностей рядків її матриці коефіцієнтів та спотворень у правій частині.

Лема 6. Нехай $\pi_{\mathbf{B}}(P_1)$ і $\pi_{\mathbf{B}}(P_2)$ – ймовірності вірного відновлення істинного розв'язку системи рівнянь (1) з використанням довільного алгоритму $\mathbf{B}$ за умови, що випадкові вектори (A_i,ξ_i), $i\in\overline{1,m}$, є незалежними в сукупності та розподілені за законами P_1 і P_2 відповідно. Тоді

$$|\pi_{\mathbf{B}}(P_1)-\pi_{\mathbf{B}}(P_2)|\le m\,d(P_1,P_2)\,. \qquad (26)$$

Доведення. Нерівність (26) випливає безпосереньо з тверджень 1 і 2 леми 3.

Наступна лема містить основні властивості випадкових векторів (25), які формуються за вхідною системою рівнянь (22).

Лема 7. Нехай виконуються умови леми 5 та рівності (19), (24). Тоді для будь-яких $\varepsilon\in(0,\theta)$ та $i\in\overline{1,m}$ з імовірністю не менше за $1-2^{1-n}-\exp\{-2n^{1+\alpha}\varepsilon^2\}$ (відносно розподілу пари (G,χ)) виконуються наступні твердження:

1) відстань по варіації між розподілом випадкового вектора

$$(A_i=G_{\mu_{1,i}}\oplus\cdots\oplus G_{\mu_{k,i}},\ \xi_i=\chi_{\mu_{1,i}}\oplus\cdots\oplus\chi_{\mu_{k,i}})$$

та розподілом ймовірностей $U_{\xi_i}(x,u)=2^{-n}\mathbf{P}\{\xi_i=u\}$, $(x,u)\in V_{n+1}$, не перевищує 2^{-cn};

2) випадкова величина ξ_i приймає значення, що дорівнює нулю, з імовірністю не менше за $\frac{1}{2}\left(1+(\theta-\varepsilon)^k\right)$.

Доведення. Позначимо $\Im_1$ та $\Im_2$ події (у просторі значень випадкового елементу (G,χ)), що полягають у невиконанні умови 1) та умови 2) відповідно. На підставі наслідку 2 і леми 5 справедлива нерівність $\mathbf{P}_{G,\chi}(\Im_1)<2^{1-n}$.

Для оцінки ймовірності події $\Im_2$ скористаємося формулою (18), у відповідності з якою

$$\mathbf{P}\{\xi_i=0\}=\frac{1}{2}\left(1+\left(T^{-1}\sum_{j=1}^{T}(-1)^{\chi_j}\right)^k\right).$$

Використовуючи лему 1, отримаємо, що

$$\mathbf{P}_{G,\chi}(\Im_2)=\mathbf{P}_{G,\chi}\left\{\mathbf{P}\{\xi_i=0\}<\frac{1}{2}\left(1+(\theta-\varepsilon)^k\right)\right\}\le$$

$$\mathbf{P}_{\chi}\left\{T^{-1}\sum_{j=1}^{T}(-1)^{\chi_j}<\theta-\varepsilon\right\}\le$$

$$\le\mathrm{P}_{\chi}\left\{T^{-1}\sum_{j=1}^{T}(-1)^{\chi_j}-T^{-1}\sum_{j=1}^{T}\theta_j<-\varepsilon\right\}\le$$

$$\le\exp\{-2T\varepsilon^2\}=\exp\{-2n^{1+\alpha}\varepsilon^2\}\,.$$

Таким чином, на підставі отриманих нерівностей

$$1-\mathrm{P}_{G,\chi}(\Im_1\cup\Im_2)\ge 1-2^{1-n}-\exp\{-2n^{1+\alpha}\varepsilon^2\}\,,$$

що й треба було довести.

Сформулюємо, нарешті, основний результат даного пункту.

Теорема 2. Нехай $\mathbf{B}$ – довільний алгоритм, що дозволяє відновлювати істинний розв'язок системи рівнянь (1), яка задовольняє умовам (2), (3) і має випадкову рівноймовірну матрицю коефіцієнтів розміру $m\times n$, з надійністю $\pi_{\mathbf{B}}(\theta)$ і трудомісткістю $T_{\mathbf{B}}$.

Розглянемо систему рівнянь (22), що задовольняє умовам (23), (24) і має випадкову рівноймовірну матрицю коефіцієнтів розміру $T\times n$, де $T=n^{1+\alpha}$, $\alpha>0$. Позначимо $\mathbf{B}'$ алгоритм розв'язання цієї системи рівнянь, який складається з наступних кроків.

1. Покласти

$$k=\left\lceil\frac{2n}{\alpha\log n}\left(\frac{1+c}{1-c}\right)\right\rceil,\ 0<c<1 \qquad (27)$$

та застосувати процедуру самокорекції з параметрами (m,k) до вхідних даних (G,h).

2. Подати отриманий список вигляду (25) на вхід алгоритму $\mathbf{B}$ та знайти за допомогою останнього шуканий вектор a.

Тоді алгоритм $\mathbf{B}'$ дозволяє відновлювати істинний розв'язок системи рівнянь (22) з ймовірністю

$$\pi_{\mathbf{B}'}(\theta)\ge\left(1-2^{1-n}-\exp\{-2n^{1+\alpha}\varepsilon^2\}\right)\times$$
$$\times\left(\pi_{\mathbf{B}}\left((\theta-\varepsilon)^k\right)-2^{-cn}m\right), \qquad (28)$$

де ε – довільне число з інтервалу $(0,\theta)$ (за умови, що обидва співмножники у правій частині формули (28) є додатними числами), використовуючи

$$T_{\mathbf{B}'}=O(T_{\mathbf{B}}+mk) \qquad (29)$$

операцій над n-вимірними двійковими векторами.

Доведення. Справедливість формули (29) випливає безпосередньо з означення алгоритму $\mathbf{B}'$. Далі, значення (27) отримується шляхом підстановки виразу (21) у формулу (19) і для доведення нерівності (28) достатньо скористатися твердженнями лем 6 та 7.

Дійсно, згідно з лемою 7, сумарна ймовірність пар (G,χ), для яких виконуються обидва твердження 1) і 2), є не менше за $1-2^{1-n}-\exp\{-2n^{1+\alpha}\varepsilon^2\}$. При цьому на підставі зазначених тверджень та нерівності (26) для кожної з цих пар ймовірність вірного відновлення вектора a на другому кроці алгоритму $\mathbf{B}'$ є не менше за $\pi_{\mathbf{B}}\left((\theta-\varepsilon)^k\right)-2^{-cn}m$.

Теорему доведено.

З наведеної теореми неважко отримати один з основних результатів роботи [6] про існування алгоритму розв'язання системи рівнянь (22) зі складністю $2^{O\left(\frac{n}{\log\log n}\right)}$.

Розглянемо в ролі **В** алгоритм розв'язання системи рівнянь (1), який будується у відповідності з доведенням теореми 1 за алгоритмом BKW з параметрами $u=\left\lceil \frac{\log\log n}{2} \right\rceil$ та $v=\left\lceil \frac{2n}{\log\log n} \right\rceil$ (див. лему 2). Нехай для простоти

$$u=\frac{\log\log n}{2},\ v=\frac{2n}{\log\log n}.$$

Тоді на підставі формул (8) та (27)

$$2r=2^u=\sqrt{\log n},$$

$$2rk=\frac{2n}{\sqrt{\log n}}\left(\frac{1+c}{1-c}\right)=O\left(\frac{n}{\sqrt{\log n}}\right).$$

Позначимо $\theta_\varepsilon=\theta-\varepsilon$ і покладемо

$$t=\left\lceil 2C\,\theta_\varepsilon^{-2rk}\ln n \right\rceil,\ \lambda=\left\lceil \ln(tn\delta^{-1}) \right\rceil,$$

вважаючи, що величини $C>1$, $\delta\in(0,1)$ і θ є константами (тобто не залежать від n). На підставі леми 2 і теореми 1 справедливі наступні нерівності:

$$\pi_{\mathbf{B}}\left(\theta_\varepsilon^{\,k}\right)\ge(1-e^{-\lambda})^{tn}(1-\exp\{-1/2\cdot\theta_\varepsilon^{\,2kr}t\})^n\ge$$

$$\ge 1-tne^{-\lambda}-n\exp\{-1/2\cdot\theta_\varepsilon^{\,2kr}t\}\ge 1-\delta-n^{1-C}.$$

Крім того,

$$m=nlt=nt(\lambda+u-1)2^v=$$

$$=O\left(n^2\sqrt{\log n}\,\log(\theta_\varepsilon^{-1})\,\theta_\varepsilon^{-2kr}2^{\frac{2n}{\log\log n}}\right)=$$

$$2^{O\left(\frac{n}{\log\log n}\right)}$$

і, отже,

$$T_{\mathbf{B}}=O(unlt)=O(um)=2^{O\left(\frac{n}{\log\log n}\right)}.$$

Підставляючи зазначені оцінки в формули (28), (29), отримаємо, що

$$\pi_{\mathbf{B}'}(\theta)\ge\left(1-2^{1-n}-\exp\{-2n^{1+\alpha}\varepsilon^2\}\right)\times$$

$$\times\left(1-\delta-n^{1-C}-2^{-cn}m\right),\qquad(30)$$

за умови, що обидва співмножники у правій частині цієї нерівності є додатними числами;

$$T_{\mathbf{B}'}=O(T_{\mathbf{B}}+mk)=2^{O\left(\frac{n}{\log\log n}\right)}.\qquad(31)$$

Таким чином, доведено наступне твердження [6].

Наслідок 3. Нехай система рівнянь (22) задовольняє умовам (23), (24) і має випадкову рівноймовірну матрицю коефіцієнтів розміру $T\times n$, де $T=n^{1+\alpha}$, $\alpha>0$. Тоді існує алгоритм, який знаходить істинний розв'язок цієї системи рівнянь з надійністю (30) і трудомісткістю (31).

ВИСНОВКИ

Найбільш трудомістким етапом суб'експоненційних алгоритмів розв'язання систем лінійних булевих рівнянь зі спотвореними правими частинами [4–6] є знаходження адитивних представлень певних булевих векторів у списку, що складається з рядків матриці коефіцієнтів заданої системи рівнянь. Для розв'язання останньої задачі (за умови незалежного, випадкового та рівноймовірного вибору зазначених рядків) можна використовувати алгоритми BKW [5], Вагнера [9] або узагальнення останнього алгоритму, запропоноване Міндером і Синклером [10].

Алгоритм Вагнера є менш трудомістким у порівнянні з алгоритмом BKW, але має ту ж саму асимптотичну часову складність, що й останній. Обидва зазначених алгоритми дозволяють розв'язувати системи з $2^{O\left(\frac{n}{\log n}\right)}$ лінійних рівнянь від n невідомих зі спотвореними правими частинами та випадковими рівноймовірними матрицями коефіцієнтів, використовуючи $2^{O\left(\frac{n}{\log n}\right)}$ операцій. Алгоритми з [5, 9, 10] базуються на ідеї одночасного виключення декількох невідомих, яка лежить в основі алгоритму Коновальцева розв'язання систем лінійних алгебраїчних рівнянь над скінченним полем. При цьому в [10] для зменшення трудомісткості алгоритму пропонується розбивати вхідні вектори на слова різної довжини (зауважимо, що аналогічна ідея використовується в [12] для побудови оптимальних у певному класі модифікацій алгоритму Коновальцева).

У випадку, коли вхідна система містить невелику кількість рівнянь або розподіл її матриці коефіцієнтів помітно відрізняється від рівномірного, можна використовувати алгоритм, наведений у формулюванні теореми 2. Зазначений алгоритм базується на ідеях робіт [6, 7] і дозволяє розв'язувати системи з $n^{1+\alpha}$ ($\alpha>0$) лінійних рівнянь від n невідомих зі спотвореними правими частинами та випадковими рівноймовірними матрицями коефіцієнтів за $2^{O\left(\frac{n}{\log\log n}\right)}$ операцій.

З практичного погляду, важливою задачею подальших досліджень є отримання оцінок надійності й трудомісткості алгоритмів розв'язання задачі про адитивне представлення [5, 9, 10] у випадку нерівномірного розподілу векторів, що утворюють вхідні списки. Цікавим є також питання про оптимальність зазначених алгоритмів. Зауважимо, що, згідно з [14], алгоритм Коновальцева є асимптотично оптимальним у класі таких алгоритмів розв'язання невироджених систем лінійних рівнянь над скінченним полем, які базуються на елементарних перетвореннях рядків їх матриць коефіцієнтів.

Литература

[1] *Балакин Г.В.* Введение в теорию случайных систем уравнений // Труды по дискретной математике. – М.: ТВП. – 1997. – Т. 1. – С. 1–18.

[2] *Левитская А.А.* Системы случайных уравнений над конечными алгебраическими структурами //

Кибернетика и системный анализ. – 2005. – Т. 41, № 1. – С. 82–116.

[3] *Berlekamp E.R.*, *McElice R.J.*, *van Tilborg H.* On the inherent intractability of certain coding problems // IEEE Trans. Inform. Theory. – 1978. – Vol. 24. – № 3. – P. 384 – 386.

[4] *Коваленко І.М.* Про алгоритм суб'експоненційної складності декодування сильно спотворених лінійних кодів // Доп. АН УРСР. Сер. А. – 1988. – № 10. – С. 16 – 17.

[5] *Blum A.*, *Kalai A.*, *Wasserman H.* Noise-tolerant learning, the parity problem, and the statistical query model // J. ACM. – 2003. – Vol. 50. – № 3. – P. 506 – 519.

[6] *Lyubashevsky V.* The pariry problem in the presence of noise, decoding random linear codes, and the subset sum problem // APPROX and RANDOM'05, Proceedings. – Springer Verlag, 2005. – P. 378 – 389.

[7] *Kopparty S.*, *Saraf S.* Local list decoding and testing of random linear codes from high-error // http://web.mit.edu/swastik/www/papers.

[8] *Bhattacharyya A.*, *Indyk P.*, *Woodruff D.P.*, *Xie N.* The complexity of linear dependence problems in vector spaces // Innovations in Computer Science – ICS 2010, Bijing, China, Jan. 7 – 9, 2011, Proceedings. – P. 496 – 508.

[9] *Wagner D.* A generalized birthday problem // Advances in Cryptology – CRYPTO'02, Proceedings. – Springer Verlag, 2002. – P. 288 – 303.

[10] *Minder L., Sinclair A.* The extended k-tree algorithm // The 19th Annual ACM-SIAM Sympposium on Discrete Algorithms, Proceedings, 2009. – P. 586 – 595.

[11] *Hoeffding W.* Probability inequalities for sums of bounded random variables // J. Amer. Statist. Assoc. – 1963. – Vol. 58. – № 301. – P. 13 – 30.

[12] *Гаврилкевич М.В.*, *Солодовников В.И.* Эффективные алгоритмы решения задач линейной алгебры над полем из двух элементов // Обозрение прикл. промышл. матем. – 1995. – Т. 2. – Вып. 3. – С. 400–437.

[13] *Johansson T.*, *Jonsson F.* Fast correlation attacks through reconstruction linear polynomials // Advances in Cryptology – CRYPTO'00, Proceedings. – Springer Verlag, 2000. – P. 300 – 315.

[14] *Глухов М.М.* О сложности решения систем линейных уравнений над конечным коммутативным цепным кольцом // Труды по дискретной математике. – М.: ФИЗМАТЛИТ. – 2002. – Т. 6. – С. 14–30.

Надійшла до редколегії 14.02.2012

Олексійчук Антон Миколайович, доктор технічних наук, професор кафедри Інституту спеціального зв'язку та захисту інформації Національного технічного університету України "КПІ". Область наукових інтересів: теоретична криптографія.

УДК 621.391:519.2

Субъэкспоненциальные алгоритмы решения систем линейных булевых уравнений с искаженными правыми частями / А.Н. Алексейчук // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 128–136.

Описана общая схема построения известных субъэкспоненциальных алгоритмов решения систем линейных булевых уравнений с искаженными правыми частями. Выделены и проанализированы важнейшие вспомогательные задачи и процедуры, используемые в указанных алгоритмах, получены неасимптотические оценки их надежности. Изложенные результаты могут быть использованы при решении ряда задач криптоанализа и теории выведывания.

Ключевые слова: система линейных уравнений с искаженными правыми частями, задача об аддитивном представлении, субъэкспоненциальный алгоритм, корреляционный криптоанализ.

Библиогр.: 14 назв.

UDC 621.391:519.2

Sub-exponential algorithms for solving systems of linear Boolean equations with noised right-hand side / A.N. Alekseichuk // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 128–136.

A general framework for constructing the known sub-exponential algorithms for solving systems of linear Boolean equations with noised right-hand side is described. Significant problems and procedures used in these algorithms are considered and analysed. The obtained results can be used in solving some problems from cryptanalysis and learning theory.

Keywords: system of linear equations with noised right-hand side, additive representation theory, sub-exponential algorithm, correlation cryptanalysis.

Ref.: 14 items.

УДК 621. 391:519.2:519.7

БЛОЧНЫЕ СИММЕТРИЧНЫЕ ШИФРЫ И МАРКОВСКИЕ ПРОЦЕССЫ

И.В. ЛИСИЦКАЯ, В.И. ДОЛГОВ

Обсуждаются известные определения Марковских шифров. Представляется уточнённый подход к их определению, основывающийся на стохастических уравнениях Марковских процессов. Показано, что в соответствии с введенным определением практически любой итеративный шифр является Марковским, в частности, SPN шифры формируют в результате зашифрования Марковские процессы первого порядка, в то время как шифры, построенные с использованием Фестель подобных схем формирования цикловых функций, создают в результате зашифрования Марковские процессы второго порядка. Уточняются некоторые определения, связанные с Марковскими шифрами.

Ключевые слова: Марковский процесс, итеративный *r*-цикловый шифр, Марковская цепь.

ВВЕДЕНИЕ

В качестве введения мы здесь напомним небольшую работу, подготовленную ещё в 1978 году [1]. В этой работе излагаются некоторые важные свойства дискретных и одновременно Марковских процессов, не освещённых в достаточной степени в литературе. Мы приведём некоторые сведения из этой работы, которые будут необходимы в дальнейшем.

В работе вводится понятие Марковского дискретного процесса *k*-того порядка. Рассматривается выборка процесса $y(t_i)$, $i=1,2\ldots,n$, заданного в виде последовательности $y_1, y_2, \ldots, y_n$ выборочных (или средних за элементарный интервал дискретности) значений исходного непрерывного процесса $y(t)$, заданного на некотором конечном интервале времени (T_1, T_2). Отмечается, что наиболее полной статистической характеристикой этого процесса является многомерный закон распределения вероятностей $P(y_1, y_2, \ldots, y_n)$ совокупности его выборочных значений. Определяется понятие Марковского процесса *k*-того порядка. Мы его здесь напомним.

Определение. *Марковским процессом k-того порядка называется процесс, условный закон распределения вероятностей выборочных значений которого для каждого значения выборки y_l, относительно предыдущих значений $y_{l-1}, y_{l-2}, \ldots, y_1$ при любом $l > k$ зависит только от k предшествующих значений, т.е.*

$$P(y_l / y_{l-1}, y_{l-2}, \ldots, y_1) = P(y_l / y_{l-1}, y_{l-2}, \ldots, y_{l-k}) .$$

Показано, что Марковский и одновременно нормальный процесс математически описывается стохастическим дифференциальным уравнением соответствующего порядка, дискретным аналогом которого является линейное неоднородное разностное уравнение со случайной правой частью [2]:

$$y_l + \sum_{p=1}^{k} a_p y_{l-p} = \eta_l \text{ , для } l > k,\ a_p = a_p^{(l)} . \qquad (1)$$

Отметим, что при $l \le k$ имеем начальные условия

$$y_l + \sum_{p=0}^{l-1} a_p^{(l)} y_{l-p} = \eta_l ,\ \ a_0^{(l)} = 0 .$$

В (1) η_l – отсчёт (выборочное значение) случайного δ-коррелированного процесса с нулевым математическим ожиданием и фиксированной дисперсией.

Нас далее будут интересовать сначала Марковские процессы первого порядка ($k = 1$), для которых

$$P(y_l /, y_{l-1}, y_{l-2}, \ldots, y_1) = P(y_l /, y_{l-1}) . \qquad (2)$$

В [1] показано, что простейший Марковский процесс первого порядка (экспоненциально коррелированный нормальный процесс) описывается стохастическим разностным уравнением:

$$y_l = -e^h y_{l-1} + \eta_l , \qquad (3)$$

где $h = \dfrac{T_0}{\tau_k}$ (T_0 – интервал дискретности, τ_k – время корреляции процесса). Ему соответствует в непрерывном времени стохастическое дифференциальное уравнение первого порядка, но нас будет интересовать именно представление, связывающее текущее значение дискретного процесса y_l с предыдущим отсчётным значением y_{l-1}.

Прежде чем идти дальше, полезно будет обобщить приведенные выше сведения следующим образом: *для Марковского процесса первого порядка соседние отсчётные значения процесса* (*два соседних значения*) *связаны между собой случайной компонентой, для Марковского процесса второго порядка три смежных отсчётных значения связаны между собой одной или более случайными компонентами, наконец, для Марковского процесса k-того порядка выборка из k + 1-го соседних отсчётных значений процесса связаны между собой случайными компонентами.*

Приведенные сведения и будут той основой, на которой мы будем строить определения для Марковских шифров.

Мы здесь хотим извиниться перед читателями за то, что статья носит больше фрагментарный, чем последовательный характер, но речь здесь идёт об уже вроде бы осознанных и понятных многим специалистам положениях, которые мы пытаемся уточнить (углубить их понимание).

1. ОБЗОР ПУБЛИКАЦИЙ ПО МАРКОВСКИМ ШИФРАМ

Напомним сначала положения, относящиеся к Марковским шифрам, которые приведены в немногочисленных публикаций в этом направлении. Основополагающей здесь, по-видимому, следует считать совместную работу Лэя, Мэсси и Марфу [3] 1991 года.

В этой работе рассматривается итеративный *r*-цикловый шифр, представленный авторами в виде рис. 1, на котором приведены необходимые нам обозначения.

Приводится такое определение Марковского шифра.

Определение. *Итеративный шифр с цикловой функцией* $Y = f\{X, Z\}$ *является Марковским шифром, если имеется групповая операция* $\otimes$, *определяющая дифференциал такая, что для всех значений* α ($\alpha \neq 0$) *и* β ($\beta \neq 0$) *условная вероятность*

$$P(\Delta Y = \beta \,|\, \Delta X = \alpha, X = \gamma)$$

является независимой от γ, *когда подключ* Z *является равномерно случайным.*

Далее приводится названная решающей теорема 2, которая объясняет, как указывают авторы, терминологию «Марковский шифр».

Теорема 2. *Если r-цикловый итеративный шифр является Марковским шифром и r цикловых ключей являются независимыми и равномерно распределёнными* (*случайными*), *то последовательность разностей* $\Delta X = \Delta Y(0)$, $\Delta Y(1), \ldots, \Delta Y(r)$ *является однородной Марковской цепью. Более того, эта Марковская цепь является стационарной, если разности* ΔX *являются равномерно распределёнными над ненулевыми элементами группы.*

В этой работе шифр с операцией, определяющей разности, рассматривается как группа, ΔX является входной разностью, а $\Delta Y(i)$, $i = 1, 2, \ldots, r$ – поцикловые выходные разности.

В качестве примера Марковского шифра приводится шифр DES. Отмечается, что для Марковского шифра с независимыми и равномерно распределёнными (случайными) цикловыми подключами вероятность *r*-цикловой дифференциальной характеристики определяется уравнением Чепмена-Колмогорова для Марковской цепи:

$$P\left(\Delta Y(1) = \beta_1, \Delta Y(2) = \beta_2, \ldots, \Delta Y(r) = \beta_r, \Delta X = \beta_0\right) = \prod_{i=1}^{r} P\left(\Delta Y(1) = \beta_1 \,|\, \Delta X = \beta_{i-1}\right).$$

Из этого следует, что вероятность *r*-циклового дифференциала (β_{01}, β_r) есть

$$P\left(\Delta Y(r) = \beta_r \,|\, \Delta X = \beta_0\right) = \sum_{\beta_1}\sum_{\beta_2}\cdots\sum_{\beta r-1}\prod_{i=1}^{r} P\left(\Delta Y(1) = \beta_1 \,|\, \Delta X = \beta_{i-1}\right),$$

где суммы рассматриваются над всеми возможными значениями разностей между различными элементами, т.е. над всеми элементами группы, исключая нейтральный элемент *e*.

Заметим здесь, что в теореме 2 и последующих разъяснениях говорится о Марковских шифрах с независимыми и равномерно распределёнными (случайными) цикловыми подключами. Как станет понятно из дальнейшего, само понятие Марковского шифра включает отмеченные выше свойства цикловых подключей, так что специальное оговаривание для Марковского шифра независимости и случайности цикловых подключей является, конечно, лишним, т.е. аккуратнее было бы говорить от том, что итеративный шифр является Марковским, если *цикловые подключи являются независимыми и равномерно распределёнными.*

Полезно будет напомнить здесь также гипотезу статистической эквивалентности, представленную авторами в рассматриваемой работе:

Гипотеза статистической эквивалентности: *Для* $(r-1)$*-циклового дифференциала* (α, β)

$$P\{\Delta Y(r-1) = \beta \,|\, \Delta X = \alpha) \approx P\{\Delta Y(r-1) = \beta \,|\, \Delta X = \alpha, Z^{(1)} = \omega_1, \ldots, Z^{(r-1)} = \omega_{r-1}\}$$

почти для всех подключевых значений $(\omega_1, \ldots, \omega_{r-1})$.

Эта гипотеза в работе используется для обоснования условий уязвимости итеративного шифра к атакам дифференциального криптоанализа. Нам она понадобится для обоснования другого факта. Мы далее покажем, что эта гипотеза выполняется для всех итеративных шифров, достигших стационарного состояния.

Приведём также теорему 3 этой работы:

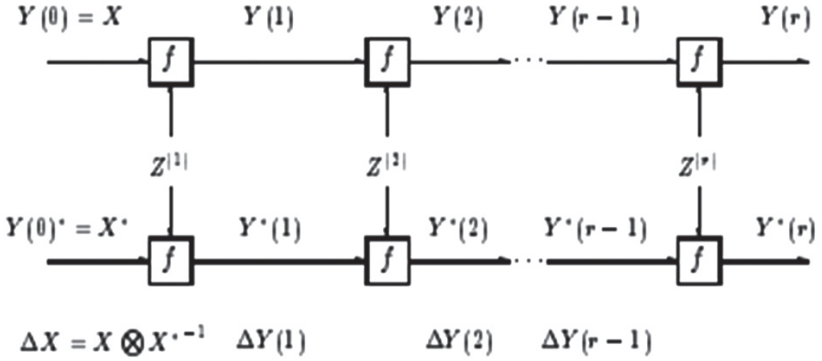


Рис. 1. Шифрование пары плайнтектов в *r*-цикловом итеративном шифре

Теорема 3. *Для Марковского шифра блочной длины m с независимыми и равномерно распределёнными цикловыми подключами, если полубесконечная Марковская цепь* $\Delta X = \Delta Y(0), \Delta Y(1), \ldots$ *имеет «равномерно-устойчивое вероятностное» распределение, т.е. существует вероятностный вектор* $(p_1, p_2, \ldots, p_M)$ *такой, что для всех* α_i

$$\lim_{r\to\infty} P\left(\Delta Y(r) = \alpha_j \mid \Delta X = \alpha_i\right) = p_j,$$

то это равномерно-устойчивое распределение должно быть равномерным $(1/M, 1/M, \ldots, 1/M)$, т.е. $\lim_{r\to\infty} P\left(\Delta Y(r) = \alpha_j \mid \Delta X = \alpha_i\right) = \frac{1}{2^m - 1}$ для каждого дифференциала (α, β), так что каждый дифференциал является, грубо говоря, равно вероятным для достаточно большого числа циклов. Если мы предполагаем дополнительно, что для этого Марковского шифра выполняется гипотеза статистической эквивалентности, то для почти всех подключей этот шифр безопасный против атаки дифференциального криптоанализа после достаточного числа циклов [3].

Мы далее покажем, что первая часть утверждения этой теоремы не соответствует реальному состоянию дел, не говоря уже о том, что гипотеза статистической эквивалентности выполняется для Марковских шифров безусловно.

Приведём также выдержки из другой работы [4], в которой затрагиваются Марковские шифры. Это работа авторов L. Keliher-a, H. Meijer-a и S. Tavares-a. Мы далее привязываемся к обозначениям именно этой работы.

В [4] *R*-цикловый шифр определяется аналитически как отображение $\varepsilon : \{0,1\}^N \to \{0,1\}^N$, для которого цикл *r* задается функцией $\mathbf{y} = \varepsilon_r\{\mathbf{x}; \mathbf{k}^r\}$; $\mathbf{x} : \{0,1\}^N$ является цикловым входом, $\mathbf{k}^r : \{0,1\}^N$ является подключом *r*-того цикла. Тогда, отмечается в этой работе, при применении для сложения с ключом групповой операции XOR ($\oplus$) над $\{0,1\}^N$ *R*-цикловый шифр ε является Марковским шифром, если для $1 \le r \le R$ и любых $\mathbf{x}, \Delta\mathbf{x}, \Delta\mathbf{y} \in \{0,1\}^N$,

$$\begin{aligned}&\mathrm{Prob}_{\mathbf{K}}\{\varepsilon_r(\mathbf{x};\mathbf{K}) \oplus \varepsilon_r(\mathbf{x} \oplus \Delta\mathbf{x};\mathbf{K}) = \Delta\mathbf{y}\} = \\ &= \mathrm{Prob}_{\mathbf{X},\mathbf{K}}\{\varepsilon_r(\mathbf{X};\mathbf{K}) \oplus \varepsilon_r(\mathbf{x} \oplus \Delta\mathbf{x};\mathbf{K}) = \Delta\mathbf{y}\},\end{aligned} \quad (4)$$

где **X** и **K** независимые случайные значения, равномерно распределенные над $\{0,1\}^N$ и **K** множество всех независимых ключей соответственно.

Соотношение (4), отмечают авторы, определяет вероятность для ключа, который фиксированное входное различие преобразует в фиксированное выходное различие, не зависящие от циклового входа.

Легко показать, отмечается также в этой работе, что SPN шифры с фиксированными S-блоками являются Марковскими шифрами.

Из представленных материалов следует, что все подходы к заданию (описанию) Марковских шифров связываются с уравнениями для дифференциалов, что, как мы покажем далее, является не совсем аккуратным. Кроме того, ряд из представленных утверждений, как видно из замечаний, представленных по тексту, представляются не совсем корректными. Мы в этой и последующей работе поставили задачу более строгого обоснования Марковских шифров и уточнения ряда принципиальных моментов, связанных с ними.

2. УТОЧНЁННЫЙ ПОДХОД К ОПРЕДЕЛЕНИЮ МАРКОВСКИХ ШИФРОВ

Первое положение, которое мы хотим здесь сначала обосновать, состоит в том, что практически любой современный шифр является шифром, формирующим в результате выполнения процедуры зашифрования Марковский процесс.

Мы здесь предлагаем свою, как нам кажется, более строгую (последовательную) точку зрения к определению Марковских шифров.

Рассмотрим более детально SPN шифр, представленный на рис. 2, заимствованным из работы [5].

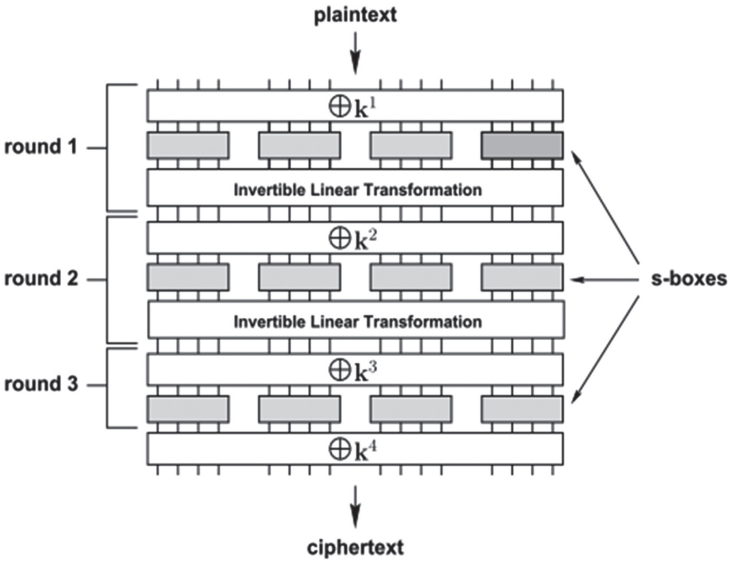


Рис. 2. SPN с $N = 16$, $M = n = 4$, и $r = 3$

Эта r-цикловая подстановочно-перестановочная схема (SPN) требует $r + 1$ N-битных подключей, $\mathbf{k}^1$, $\mathbf{k}^2$, ..., $\mathbf{k}^r$, $\mathbf{k}^{r+1}$. Каждый цикл (раунд) состоит из трёх стадий, или слоёв.

В ключесмешивающей стадии N-битный цикловой выход является *побитным XOR-ом (суммой по модулю два) с подключом для этого цикла.*

В подстановочной стадии результирующий блок делится на M подблоков размера n ($N = Mn$), и каждый подблок становится входом в биективный $n \times n$ подстановочный блок (S-блок) – являющийся биективным отображением из $\{0,1\}^n$ в $\{0,1\}^n$.

В стадии линейного преобразования, выход подстановочной стадии обрабатывается инвертируемым N-битным линейным преобразованием (классическое линейное преобразование было поразрядной перестановкой, откуда и следует появление названия подстановочно-перестановочная схема [6]). Линейное преобразование обычно исключается из последнего цикла, так как легко показать, что его включение в преобразование не добавляет стойкости шифру. Финальный подключ $\mathbf{k}^{r+1}$ XOR-рируется (суммируется по модулю 2) с выходом цикла r чтобы сформировать шифртекст. Предполагается, что те же самые линейные преобразования используются в каждом цикле. Если не оговорено иного, то никаких ограничений не установлено и на выбор S-блоков.

Расшифрование выполняется прогоном SPN «обратно (задом наперед)». Подключ $\mathbf{k}^{r+1}$ сначала XOR-рируется с зашифрованным текстом, и затем в каждом цикле r (из R вплоть до 1-го), выполняется обратное линейное преобразование, сопровождаемое обратными S-блоками, и результирующий блок XOR-рируется с $\mathbf{k}^1$.

Здесь мы подходим к тому, что если выделить в рассмотренном шифре сложение с цикловым подключом в отдельное преобразование (это можно сделать практически в любом SPN шифре), то одноцикловое преобразование такого шифра всегда можно представить в виде результата выполнения над входом в цикловую функцию преобразования F (прохождение через S-блоки и линейное преобразование) и последующего сложения результата преобразования со случайной компонентой, определяемой цикловым подключом, т.е. блок данных на выходе цикловой функции будет иметь вид:

$$\mathbf{y} = \varepsilon_r\{\mathbf{x}; \mathbf{k}^{r+1}\} = F_r\{\mathbf{x}\} + \mathbf{k}^{r+1}. \qquad (5)$$

В результате мы приходим к аналогу уравнения (3), особенностью которого является то, что в уравнении (5) над цикловым входом $\mathbf{x}: \{0,1\}^N$ выполняется преобразование не линейного типа, как это сделано в уравнении (3), а нелинейное преобразование, которое осуществляется в поле $\mathbf{x}, \mathbf{y} \in \{0,1\}^N$. Но это всё равно получается уравнение, связывающее предыдущее значение входа $\mathbf{x}$ с текущим $\mathbf{y}$ (это уже новый вход в очередной цикл) с помощью случайной компоненты $\mathbf{k}^{r+1}$. Если полагать, что ключи для циклов выбираются равновероятно и независимо, то это уравнение Марковского процесса (теперь, конечно же, отличающегося от нормального). В результате предлагается определение Марковского шифра в виде:

Определение 1. *Марковским шифром* (*первого порядка*) *является* (*называется*) *шифр, для которого соотношение для выхода цикловой функции с её входом для любого значения входа и выхода* $\mathbf{x}, \mathbf{y} \in \{0,1\}^N$ *определяется нелинейным уравнением* (5), *в котором случайная компонента в правой части является ключевым значением* $\mathbf{k}^{r+1} \in \{0,1\}^N$, *выбранным независимо и равновероятно из всего множества возможных ключей.*

Можно ввести и определение Марковского шифра k-того порядка:

Определение 1′. *Марковским шифром* (*k-того порядка*) *называется шифр, для которого соседние значения выходов k цикловых функций и значение входа в первую цикловую функцию их этого набора связаны между собой нелинейно случайной компонентой* (*нелинейное уравнение, связывающее $k + 1$ соседних значений на выходах цикловых функций, содержит случайную компоненту*).

В этом случае мы имеем в виду уравнение, которое отличается от (1) нелинейной связью переменных, входящих в него.

Мы здесь привели определение Марковского шифра k-того порядка, так как нам потребуется в рамках этой работы и Марковские шифры второго порядка.

Установим (определим) теперь связь приведенных определений с определениями, известными из литературы.

Для входа в цикловую функцию $\mathbf{x}' = \mathbf{x} \oplus \Delta\mathbf{x}$, с учётом (5) имеем:

$$\mathbf{y}' = \varepsilon_r\{\mathbf{x} \oplus \Delta\mathbf{x}; \mathbf{k}^{r+1}\} = F_r\{\mathbf{x}'\} + \mathbf{k}^{r+1}.$$

В результате для дифференциалов (разностей) циклового преобразования $\mathbf{y} \oplus \mathbf{y}' = \Delta\mathbf{y}_r$, $\mathbf{x} \oplus \mathbf{x}' = \Delta\mathbf{x} = \Delta\mathbf{y}_{r-1}$ для одного и того же ключевого значения $\mathbf{k}^{r+1}$ приходим к уравнению

$$\Delta\mathbf{y}_r = F_r\{\mathbf{x}\} \oplus F_r\{\mathbf{x}'\} = F_r^*\{\Delta\mathbf{x}\} = F_r^*\{\Delta\mathbf{y}_{r-1}\}. \qquad (6)$$

В (6) F_r^* – функция циклового преобразования разностей. В рассмотренных выше и других публикациях [3-8] понятие Марковского шифра связывают именно с уравнением для дифференциалов. Ещё один пример. В [7] Марковским назван шифр, у которого уравнение шифрования на одном цикле удовлетворяет условию: вероятность дифференциала не зависит от выбора открытых текстов. Тогда, если подключи циклов между собой независимы, то последовательность разностей после каждого цикла образует Марковскую цепь, где последующее состояние определяется только предыдущим.

Конечно, это и приведенные выше понятия согласуются с отмеченным определением

Марковского процесса первого порядка для дифференциалов.

Но мы хотим сейчас привлечь внимание к имеющимся в литературе подходам к делению шифров на Марковские и немарковские. Нам представляется, что пропущенная многими авторами связь, выражаемая в виде уравнения, которое оперирует не с дифференциалами, а с соседними значениями одноцикловых переходов шифра, привела к не совсем аккуратной интерпретации свойств некоторых криптографических преобразований.

Так в [8] шифр ГОСТ 28147-89 относится к немарковским. Напомним, что в режиме простой замены шифра ГОСТ-а [9] 64-битный блок открытого текста (сообщения) разбивается на две части по 32 бита каждая (правая половина блока далее обозначена A_0, а левая B_0). Осуществляется 32 однотипных цикла преобразования, структура которых в каждом из циклов описывается выражениями

$$A_i = f(A_{i-1} \left[+\right] K_j) \oplus B_{i-1}, \qquad (7)$$

$$B_i = A_{i-1}, \qquad (8)$$

причем для $i = \overline{1,24}$ берется $j = (i-1) \bmod 8$, для $i = \overline{25,31}$ соответственно $j = 32 - i$, и для последнего цикла

$$A_{32} = A_{31},$$

$$B_{32} = f(A_{31} \left[+\right] K_0) \oplus B_{31},$$

где i – номер итерации; символом $\left[+\right]$ обозначена операция сложения по модулю 2^{32}. Тогда с учётом (8) соотношение (7) можно переписать в виде:

$$A_i = f(A_{i-1} \left[+\right] K_j) \oplus A_{i-2}.$$

Легко убедиться, что по нашему второму определению мы пришли к уравнению Марковского процесса второго порядка (если считать цикловые подключи независимыми), и, следовательно, шифр ГОСТ 28147-89 тоже является Марковским (правда, здесь случайная компонента вошла в нелинейное преобразование). Представляется, что реально существующая корреляция подключей шифра существенно не изменит картину.

Марковским шифром второго порядка является также и шифр DES, для которого уравнения зашифрования (R – правый полублок, L – левый полублок) имеют вид [9]:

$$R_i = f\left(R_{i-1}, K_i\right) \oplus R_{i-2},$$

$$L_i = R_{i-1}.$$

Для $R'_i = f\left(R'_{i-1}, K_i\right) \oplus R'_{i-2}$, где $R'_i = \Delta R \oplus R_i$ имеем:

$$\Delta R_i = f^*\left(\Delta R_{i-1}\right) \oplus \Delta R_{i-2}. \qquad (9)$$

В работе [3] шифр DES причисляется к Марковским шифрам (первого порядка). На самом же деле, это справедливо лишь в том случае, если речь идёт о дифференциальной характеристике, использованной Э. Бихамом и А. Шамиром [10] при построении предложенной ими атаки на шифр, вероятность которой (характеристики) действительно приводится к произведению вероятностей цикловых переходов, характерному для частных дифференциалов Марковских шифров первого порядка. Напомним, что при построении своей атаки они воспользовались трёхблочными характеристиками обнуляющего типа (d = 1960 0000), для которых в (9) надо положить

$$\Delta R_{i-2} = \Delta R_{i-4} = \ldots = \Delta R_{i-2k} = 0,$$

при этом, естественно, что

$$f^*\left(\Delta R_{i-2}\right) = f^*\left(\Delta R_{i-4}\right) = \ldots = f^*\left(\Delta R_{i-2k}\right) = 0.$$

Для характеристик обнуляющего типа соответственно $\Delta R_{i-1} = \Delta R_{i-3} = \ldots = \Delta R_{i-2k-1} = d$, в то время как $f^*\left(\Delta R_{i-1}\right) = f^*\left(\Delta R_{i-3}\right) = \ldots = f^*\left(\Delta R_{i-2k-1}\right) = 0$.

В результате:

$$\Delta R_i = f^*\left(\Delta R_{i-1}\right) = 0 \text{ с вероятностью } p,$$

$$\Delta R_{i-1} = f^*\left(\Delta R_{i-2}\right) \oplus \Delta R_{i-3} \rightarrow \Delta R_{i-1} = \Delta R_{i-3} = d$$
с вероятностью 1,

$$\Delta R_{i-2} = f^*\left(\Delta R_{i-3}\right) = 0 \text{ с вероятностью } p,$$

$$\Delta R_{i-3} = \Delta R_{i-5} = d \text{, с вероятностью } 1 \ldots,$$

т.е. в этом случае мы действительно приходим скорее не к уравнениям дифференциалов Марковского шифра первого порядка, а к результирующей вероятности дифференциальной характеристики выражаемой в виде произведения вероятностей однотипных цикловых переходов (рассматривается частная дифференциальная характеристика), причём половина из них происходят без снижения вероятности (с вероятностью единица).

Но самое интересное, так это то, что и Марковские шифры первого порядка и Марковские шифры второго порядка приходят к одному и тому же стационарному состоянию: таблицы полных дифференциалов и линейных корпусов шифров асимптотически повторяют законы распределения вероятностей XOR переходов и смещений таблиц линейных аппроксимаций случайных подстановок соответствующей степени. Но об этом в следующей публикации.

Возвращаясь к теореме 2, мы здесь хотим обратить внимание на присутствующее в её формулировке требование независимости дифференциалов от выбора открытых текстов (ΔX являются равномерно распределёнными над ненулевыми элементами группы) и утверждение в теореме о стационарности Марковской цепи.

Так вот, если рассматривать теорию Марковских процессов [1 и др.], то любой случайный процесс имеет своё начало и конец. Например, простейший нормальный Марковский процесс

(3) не сразу приобретает показатели стационарного распределения. Начальное его значения следует рассматривать как нестационарное. Для Марковского процесса k-того порядка (см. начальные условия для уравнения (1)) будет уже переходный процесс из k смежных значений (для нормального процесса).

Так и в шифрах. На основе многочисленных результатов проведенных экспериментов мы здесь утверждаем, что большинство современных шифров (использующих для введения циклового подключа не только групповую операцию XOR) являются Марковскими (текущее значение шифртекста является функцией конечного числа предыдущих значений шифртекстов). Для каждого такого шифра существует определённое (небольшое) число начальных циклов, после которого законы распределения переходов XOR таблиц (дифференциалов) приходят к установившемуся (стационарному) значению. На первых шагах шифрования Марковскому шифру присущ переходный период, который вполне согласуется с положениями теории Марковских процессов. Шифр приходит к стационарному процессу (состоянию) асимптотически.

И ещё в отношении равномерности распределения входных разностей ΔX над ненулевыми элементами группы, отмеченными в теореме 2. Дело в том, что уравнение (5) и соответствующее ему уравнение (6) определяют Марковский процесс при любом значении входа $\mathbf{x} \in \{0,1\}^N$. Множество равновероятных значений входов необходимо лишь для формирования закона распределения переходов XOR таблицы шифра, повторяющего распределение дифференциалов случайной подстановки, которое устанавливается после переходного периода.

Остаётся отметить, что после публикации линейного криптоанализа, теория Марковских шифров была распространена на сопротивляемость линейному криптоанализу, что привело к аналогичным выводам для линейных приближений (корпусов) шифров [11]. Для нас в этом нет ничего нового. Наши исследования [12,13 и др.] и в этом случае свидетельствуют, что линейные показатели шифров при росте числа циклов приходят к соответствующим показателям случайных подстановок (Марковский шифр приходит к стационарному состоянию и по этому показателю). Напомним здесь, что равенства, используемые при построении линейных аппроксимаций шифров, определяют связь соседних значений входов и выходов цикловых функций (прошедших соответствующие маски) через случайную компоненту (сумму ключевых битов).

Здесь мы хотим остановиться, хотя у нас есть ещё претензии к использованию матриц переходных вероятностей при оценке показателей стойкости шифрующих преобразований (Марковских шифров), да и в целом к развиваемой во многих работах самой методике оценки стойкости БСШ к атакам дифференциального и линейного криптоанализа. Мы на них остановимся в нашей следующей работе.

ВЫВОДЫ

К основным выводам из представленных в работе результатов, предложений и соображений можно отнести такие:

1. Имеющиеся в литературе подходы к определению Марковских шифров представляются не совсем аккуратными.

2. Предлагается уточнённый подход к определению Марковских шифров, который строится на основе строгого математического определения Марковского случайного процесса k-того порядка.

3. Показано, что в соответствии с введенным определением практически любой итеративный шифр является Марковским, в частности, SPN шифры формируют в результате зашифрования Марковские процессы первого порядка, в то время как шифры, построенные с использованием Фестель подобных схем формирования цикловых функций, создают в результате зашифрования Марковские процессы второго порядка.

4. Для каждого итеративного (Марковского) шифра существует определённое (небольшое) число начальных циклов шифрования, после которого законы распределения переходов XOR таблиц (дифференциалов) и смещений таблиц линейных аппроксимаций (линейных корпусов) шифра приходят к установившемуся (стационарному) значению. На первых шагах шифрования Марковскому шифру присущ переходный период. Шифр приходит к стационарному процессу (состоянию) асимптотически.

Литература

[1] *Долгов В.И.* Вопросы теории и цифрового моделирования нормальных Марковских процессов. / МО. – 1978. – 74 с.

[2] *Иванов В.А.* Математические основы теории автоматического регулирования / В.А. Иванов, Б.К.Чемоданов, В.С. Медведев // Изд-во «Высшая школа». – М., 1971. – 755 с.

[3] *X. Lai, J. Massey, and S. Murphy,* Markov ciphers and differentjal cryptanalysis, Advances in Cryptalogy – EUROCRYPT'93, LNCS 547, Springer-Verlag, pp. 17-38, 1991.

[4] *L. Keliher, H. Meijer, and S. Tavares,* Toward the true random cipher: On expected linear probability values for SPNs with randomly selected s-boxes, chapter in Communications, Information and Network Security, V. Bhargava, H. Poor, V. Tarokh, and S. Yoon (Eds.), pp. 123-146, Kluwer Academic Publishers, 2003.

[5] *F. Sano, K. Ohkuma, H. Shimizu, S. Kawamura.* On the Security of Nested SPN Cipher against the Differential and Linear Cryptanalysis/ IEICE Trans. Fundamentals, vol. E86-a, NO.1 January 2003, pp. 37-46.

[6] *H. Feistel*, Cryptography and computer privacy, Scientific American, Vol. 228, No. 5, pp. 15–23, May 1973.

[7] *L. Keliher,* Linear Cryptanalysis of Substitution-Permutation Networks. A thesis submitted to the School of Computing in conformity with the requirements for the degree of Doctor of Philosophy, 2003, 160 p.

[8] *Ковальчук Л.В.* Сходимость последовательности матриц вероятностей дифференциальных аппроксимаций немарковского блочного шифра к равновероятной матрице при увеличении количества циклов. // Прикладная радиоэлектроника – 2007. – Т.5, № 2 – С. 274-276.

[9] ГОСТ 28147-89. Системы обработки информации. Защита криптографическая. Алгоритм криптографического преобразования.

[10] *E. Biham and A. Shamir.* Differential Cryptanalysis of DES-like Cryptosystems. Journal of Cryptology, 4(1): 3-72, 1991.

[11] Joan Daemen, Vincent Rijmen Probability distributions of Correlation and Differentials in Block Ciphers. / Joan Daemen, Vincent Rijmen // April 13, 2006, pp. 1–38.

[12] *Горбенко І.Д.* Новая идеология оценки стойкости блочных симметричных шифров к атакам дифференциального и линейного криптоанализа. / І.Д. Горбенко, В.И. Долгов, И.В. Лисицкая, Р.В. Олейников // Прикладная радиоэлектроника. – 2010. – Т. 9, № 3. – С. 212-320.

[13] *Кузнецов А.А.* Линейные свойства блочных симметричных шифров, представленных на украинский конкурс. / А.А. Кузнецов, И.В. Лисицкая, С.А. Исаев // Прикладная радиоэлектроника. – 2011. Т. 10, № 2. – С. 135-140.

Поступила в редколлегию 20.02.2012

Лисицкая Ирина Викторовна, кандидат технических наук, доцент кафедры БИТ ХНУРЭ. Область научных интересов: криптография, теория сложности.

Долгов Виктор Иванович, доктор технических наук, профессор кафедры БИТ ХНУРЭ. Область научных интересов: математические методы защиты информации.

УДК 621. 391:519.2:519.7

Блочні симетричні шифри та марковські процеси / І.В. Лисицька, В.І. Долгов // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 137–143.

Обговорюються відомі визначення Марківських шифрів. Наводиться уточнений підхід до їх визначення, що ґрунтується на стохастичних рівняннях Марківських процесів. Показано, що відповідно до введених визначень практично будь-який ітеративний шифр є Марківським, зокрема, SPN шифри формують в результаті зашифрування Марківські процеси першого порядку, в той час як шифри, побудовані з використанням Фестель подібних схем формування циклових функцій, створюють в результаті зашифрування Марківські процеси другого порядку. Уточнюються деякі визначення, пов'язані з Марківськими шифрами.

Ключові слова: Марківский процес; ітеративний *r*-цикловий шифр; Марківський ланцюг.

Іл. 2. Бібліогр. 13 найм.

UDC 621. 391:519.2:519.7

Block symmetric ciphers and Markov processes / I.V. Lysytska, V.I. Dolgov // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 137–143.

The paper discusses the famous definitions of Markov ciphers and provides an updated approach to their definition based on the stochastic equations of Markov processes. It is shown that in accordance with the definition introduced almost any iterative cipher is a Markov one, in particular, SPN ciphers form Markovian first order processes as a result of encoding, while the ciphers, constructed with use of Festel-like schemes of forming cyclic functions, form Markov second order processes as a result of encoding. Some definitions, related to the Markov ciphers, are particularized.

Keywords: Markov process, iterative *r*-round cipher, Markov chain.

Fig. 02. Ref. 13 items.

УДК 621. 391:519.2:519.7

ОЦЕНКИ МАКСИМАЛЬНЫХ ЗНАЧЕНИЙ ДИФФЕРЕНЦИАЛОВ И ЛИНЕЙНЫХ КОРПУСОВ МАРКОВСКИХ ШИФРОВ

В.И. ДОЛГОВ, И.В. ЛИСИЦКАЯ, А.А. НАСТЕНКО, К.Е. ЛИСИЦКИЙ

Представляется краткий анализ существа известных подходов к оценке показателей доказуемой стойкости марковских блочных симметричных шифров к атакам дифференциального и линейного криптоанализа. Излагается сущность новой методологии оценки доказуемой стойкости БСШ к атакам дифференциального и линейного криптоанализа. Приводятся результаты оценки максимальных значений дифференциалов и линейных корпусов Марковских шифров. Выполняется обоснование процесса перехода шифров к стационарному состоянию. Формулируется новая редакция гипотезы статистической эквивалентности. Уточняются некоторые положения теории Марковских шифров, встречающиеся в публикациях.

Ключевые слова: Марковский шифр; дифференциальная вероятность; линейная вероятность; новая методология оценки доказуемой стойкости; стационарное состояние свойственное случайной подстановке.

ВВЕДЕНИЕ

В нашей предыдущей работе [1] был поднят вопрос об уточнении ряда принципиальных моментов, связанных с понятиями и определениями теории Марковских шифров. Было введено понятие Марковского шифра k-того порядка и сделан общий вывод о том, что все известные итеративные блочные шифры являются Марковскими шифрами (первого или второго порядка). В этой работе мы продолжаем обсуждение состояния ряда теоретических и практических вопросов в этом направлении, и здесь мы хотим привлечь внимание к подходам, имеющимся в публикациях, посвящённым оценкам максимальных значений дифференциалов и линейных корпусов Марковских шифров. Мы изложим сущность новой методологии формирования показателей доказуемой стойкости блочных симметричных шифров, развиваемой на кафедре БИТ ХНУРЭ, которая естественно применима и к Марковским шифрам, и покажем, что соответствующие показатели доказуемой стойкости для таких шифров могут быть получены расчётным путём.

1. ПОНЯТИЙНЫЙ АППАРАТ ЛИНЕЙНОГО И ДИФФЕРЕНЦИАЛЬНОГО КРИПТОАНАЛИЗА

Уместно будет напомнить основной понятийный аппарат линейного и дифференциального криптоанализа, которым мы воспользуемся в дальнейшем. Следуя работе [2], приведём ряд определений.

Определение 1. (Дифференциальная и Линейная вероятности): *Дифференциальная вероятность DP^f и линейная вероятность LP^f соответственно для ключезависимой функции f с n-битным входом x и n-битным выходом y ($x, y \in GF(2^n)$) есть*

$$DP^f(\Delta x \to \Delta y) = \frac{\#\{x \in GF(2)^n \mid f(x) \oplus f(x \oplus \Delta x) = \Delta y}{2^n};$$

$$LP^f(\Gamma x \to \Gamma y) = \left(\frac{\#\{x \in GF(2)^n \mid x \cdot \Gamma x) = f(x) \cdot \Gamma y}{2^{n-1}} - 1 \right)^2,$$

где Δx и Δy являются входной и выходной разностями, а Γx и Γy – это входная и выходная маски; $x \cdot \Gamma x$ обозначает результат скалярного произведения x на Γx, $f(x) \cdot \Gamma y$ – результат скалярного произведения $f(x)$ на Γy/.

Определение 2. ($DP^f_{\max}$ и $LP^f_{\max}$): *Максимальное значение дифференциальной и линейной вероятностей для ключезависимой функции f определяются соответственно как*

$$DP^f_{\max} = \max_{\Delta x \neq 0, \Delta y} DP^f(\Delta x \to \Delta y),$$

$$LP^f_{\max} = \max_{\Gamma x,\, \Gamma x \neq 0,} LP^{f[k]}(\Gamma x \to \Gamma y).$$

Напомним также выражения для средних вероятностей *ADP*, *ALHP*, *MADP* і *MALHP* ключезависимой функции $f = f[k](x)$ с n-битным входом x и n-битным выходом, параметризованной ключом k, которые используются во многих публикациях по обоснованию показателей стойкости блочных шифров.

Определение 3. *Среднее значение дифференциальной вероятности (ADP) функции $f[k](x)$ есть*

$$ADP^f = \underset{k}{ave}\, DP^{f[k]}(\Delta x \to \Delta y).$$

Определение 4. *Среднее значение вероятности линейного корпуса (ALHP) функции $f = f[k](x)$ есть*

$$ALHP^f = \underset{k}{ave}\, LP^{f[k]}(\Gamma x \to \Gamma y).$$

Определение 5. *Максимум среднего значения дифференциальной вероятности (MADP) и максимум среднего значения вероятности линейного корпуса (MALHP) функции $f = f[k](x)$ есть*

$$MADP^f = \max_{\Delta x \neq 0, \Delta y} ADP^f(\Delta x \to \Delta y),$$

$$MALHP^f = \max_{\Gamma x,\, \Gamma y \neq 0} ALHP^f(\Gamma x \to \Gamma y).$$

Заметим теперь, что приведенные здесь определения *MADP* и *MALHP*, повсеместно используемые в публикациях, на наш взгляд не являются

адекватными задаче оценке потенциальных характеристик стойкости шифра к атакам дифференциального и линейного криптоанализа (они характеризуют лишь максимумы средних значений таких вероятностей, вычисленных для некоторого фиксированного перехода $\Delta x \to \Delta y$ или некоторого фиксированного сочетания масок $Гx \to Гy$).

Нами в [3] предложено определять не максимумы средних значений переходов и смещений, а средние (по множеству ключей) значения максимумов дифференциальных и линейных вероятностей – *AMDP* и *AMLHP* соответственно, которые определяются следующим образом:

Определение 6 (*AMDP*). *Среднее* (*по множеству из* 2^h *ключей*) *значение максимальных дифференциальных вероятностей ключезависимой функции* $f[k](x)$ *есть*

$$AMDP^f = \underset{k}{ave}\, DP_{\max}^{f[k]} = \frac{1}{2^h}\sum_{k=1}^{2^h} DP_{\max}^{f[k]} .$$

Определение 7 (*AMPLH*). *Среднее* (*по ключам*) *значение максимальных вероятностей линейных корпусов функции* $f[k](x)$ *есть*

$$AMLHP^f = \underset{k}{ave}\, LP_{\max}^{f}(Гx \to Гy) = \frac{1}{2^h}\sum_{k=1}^{2^h} LP_{\max}^{f[k]} .$$

В обоих случаях 2^h *– мощность используемого при вычислениях множества ключей зашифрования.*

Тут же можно отметить, что очевидны неравенства:

$$MADP^f < AMDP^f,$$
$$MALHP < AMLHP,$$

которые лишний раз свидетельствуют в пользу вновь введенных определений, не говоря уже о значительных вычислительных преимуществах предлагаемого подхода и полного согласования его результатов с соответствующими свойствами случайных подстановок.

Именно с использованием этих двух показателей выполнено обоснование новой идеологии оценки доказуемой стойкости БСШ в наших работах [4-7, и др.].

2. КРАТКИЙ АНАЛИЗ СУЩЕСТВА ИЗВЕСТНЫХ ПОДХОДОВ К ОЦЕНКЕ ПОКАЗАТЕЛЕЙ ДОКАЗУЕМОЙ СТОЙКОСТИ МАРКОВСКИХ БЛОЧНЫХ СИММЕТРИЧНЫХ ШИФРОВ К АТАКАМ ДИФФЕРЕНЦИАЛЬНОГО И ЛИНЕЙНОГО КРИПТОАНАЛИЗА

Достаточно детальный анализ широко эксплуатируемой сегодня концепции оценки показателей доказуемой стойкости блочных симметричных шифров представлен в нашей работе [4]. Мы здесь приведём уже сами выводы, следующие из этого анализа.

Первый вывод состоит в том, что в основе всех подходов к оценке показателей стойкости блочных симметричных шифров к атакам дифференциального и линейного криптоанализа лежит процедура определения максимумов средних значений вероятностей (максимальных средних значений вероятностей) полного дифференциала (*MADP*) для всего шифра и смещения его линейного корпуса (*MALHP*).

Добавим здесь второй вывод, который состоит в том, что все развиваемые подходы ориентированы на формирование оценочных (граничных) значений соответствующих показателей.

Третий вывод состоит в том, что оценки соответствующих показателей отличаются в значительных пределах.

Четвёртый вывод состоит в том, что результирующие показатели стойкости шифров практически во всех работах связываются с соответствующими криптографическими показателями, входящих в шифры S блоковых конструкций.

В работе [4] делается обобщающее заключение, что существующие методики оценки показателей стойкости БСШ являются все еще далеко не совершенными.

Следует сказать, что недалеко от изложенного общего состояния и движения в отмеченном направлении находятся методы и подходы, используемые при формировании показателей стойкости Марковских шифров.

Мы здесь, однако, отметим две работы, в которых оценки стойкости Марковских шифров строятся на основе использования матриц переходных вероятностей одноцикловых преобразований.

В работе [9] авторы оперируют матрицами переходных вероятностей для цепей Маркова и рассматривают результат многоциклового преобразования с помощью возведения матрицы переходных вероятностей одноциклового преобразования в степень, равную числу циклов преобразования.

В работе [10] также считается справедливой сходимость последовательности матриц вероятностей дифференциальных аппроксимаций марковского блочного шифра к равновероятной матрице при увеличении количества циклов. И в ряде других работ матрица дифференциальных вероятностей шифра строится на возведении матрицы переходных вероятностей одноциклового преобразования в степень, равную числу циклов (раундов). Это, конечно, и в первом и во втором случаях не верно, как и не верно утверждение о сходимости последовательности матриц средних вероятностей шифра к равновероятной.

Возведение в степень справедливо лишь в том случае, когда над отсчётными значениями выборки, входящими в уравнение Марковского процесса, осуществляется линейное преобразование. Когда речь идёт о полных дифференциалах, то для практически всех известных итеративных шифров этого делать нельзя, так такие шифры цикловое преобразование строят с использованием нелинейных операций (*S*-блоков). Заметим, что шифры без *S*-блоков

тоже укладываются в эту общую схему (там тоже используются нелинейные операции).

Остановимся на этом принципиальном моменте более детально.

Начнём с Марковского процесса первого порядка, заданного уравнением для дифференциалов (см. работу [1]):

$$\Delta\mathbf{y}_r = F_r\{\mathbf{x}\} \oplus F_r\{\mathbf{x}^*\} = F_r^*\{\Delta\mathbf{x}\} = F_r^*\{\Delta\mathbf{y}_{r-1}\}, \quad (1)$$

В (1) F_r^* – функция циклового преобразования разностей.

Как следует из уравнения (1), при вычислении выходной разности $\Delta\mathbf{y}_r$ ключ последнего цикла $\mathbf{k}^{r+1}$, как и ключи предшествующих циклов вроде бы де уходят из уравнения (компенсируется). Однако это не так. Ключи текущего и предшествующих циклов шифрования (случайные компоненты) присутствуют в этом преобразовании через вполне определённые значения промежуточных разностей, участвующих в формировании разности шифртекстов на его выходе. И для выходной разности это, конечно, ключезависимый результат (в том смысле, что для каждого значения ключа будет формироваться своя разность).

Тем не менее, решение уравнения (1) для фиксированного набора цикловых подключей можно представить в виде $\mathbf{y}_r \oplus \mathbf{y}_r^* =$

$$= y_r \oplus y_r^* = F_r\{\mathbf{C}_{r-1}\} \oplus \mathbf{k}^{r+1} \oplus F_r\{\mathbf{C}_{r-1}^*\} + \mathbf{k}^{r+1} =$$
$$= F_r\{\mathbf{C}_{r-1}\} \oplus F_r\{\mathbf{C}_{r-1}^*\} = F_r^*\{\Delta\mathbf{C}_{r-1}\}$$

и далее

$$F_r^*\{\Delta\mathbf{y}_{r-1}\} = F_r\{\mathbf{y}_{r-1}\} \oplus F_r\{\mathbf{y}_{r-1}^*\} =$$
$$= F_r\{F_{r-1}\{\mathbf{y}_{r-2} \oplus \mathbf{k}^r\}\} \oplus F_r\{F_{r-2}\{\mathbf{y}_{r-1}^*\} \oplus \mathbf{k}^r\} =$$
$$= F_r^*\{F_{r-1}^*\{\mathbf{y}_{r-2} \oplus \mathbf{k}^r \oplus F_{r-1}\{\mathbf{y}_{r-2}^*\} \oplus \mathbf{k}^r\} =$$
$$= F_r^*\{F_{r-1}^*\{\Delta\mathbf{y}_{r-2}\}\} = \ldots = F_r^*\{F_{r-1}^*\{\cdots F_1^*\{\Delta\mathbf{y}_0\}\cdots\},$$

т.е.

$$\Delta\mathrm{y}_r = F_r^*\{F_{r-1}^*\{F_{r-2}^*\{\ldots\{F_1^*\{\Delta\mathrm{x}_0\}\}\ldots\}, \ \Delta\mathrm{x}_0 = \Delta\mathrm{y}_0. \quad (2)$$

Можно далее ввести в рассмотрение полное множество значений входных и выходных разностей и тогда произведение преобразований $F_r^* \cdot F_{r-1}^* \cdot F_{r-2}^* \cdot \ldots \cdot F_1^*$ рассматривать как матрицу переходных вероятностей, например в виде:

$$F_r^* \cdot F_{r-1}^* \cdot F_{r-2}^* \cdot \ldots \cdot F_1^* =$$

$$= \begin{vmatrix} f_{0,0} & f_{0,1} & \cdots & f_{0,2^{n-1}} \\ f_{1,0} & f_{1,1} & \cdots & f_{1,2^{n-1}} \\ \vdots & \vdots & \vdots & \vdots \\ f_{2^{n-1},0} & f_{2^{n-1},1} & \cdots & f_{2^{n-1},2^{n-1}} \end{vmatrix}.$$

В этой матрице каждый элемент $f_{i,j}$ определяет вероятность перехода i-той разности $\Delta\mathbf{x}_i$ на входе шифра в j-тую выходную разность $\Delta\mathbf{y}_j$ (в данном случае для r циклов зашифрования). Для марковского шифра, достигшего стационарного состояния, значения элементов матрицы подчиняются закону распределения вероятностей переходов случайной подстановки соответствующей степени (матрица имеет вполне определённое число нулей, число двоек, четвёрок и т.д.). Наращивание числа циклов в этом случае приводит к матрице с другим размещением элементов, но для неё всё равно сохраняется закон распределения переходов (она снова имеет прежнее число нулей, двоек, четвёрок и т.д., вплоть до единственного, как правило, максимального значения для определённого выхода).

Остаётся отметить, что для каждого числа циклов и каждого ключа зашифрования будет своя матрица переходных вероятностей (нормированная таблица дифференциальных разностей) и результирующая матрица переходных вероятностей для r циклового Марковского шифра будет определяться, как это следует из (2), на основе произведения подстановочных преобразований для одноцикловых переходов.

Большинство известных итеративных шифров используют однотипные цикловые преобразования. Поэтому результирующее подстановочное преобразование для таких шифров будет действительно представлять собой соответствующую степень одноциклового преобразования. Здесь надо помнить, что произведение подстановочных преобразований есть последовательное их выполнение одного за другим. Таким образом, мы пришли к последовательному выполнению r однотипных преобразований:

$$F_r^* = (F_1^*)^r, \quad (3)$$

но не к матричному возведению в степень.

Следовательно, попытки использования для вычисления оценок показателей стойкости Марковских шифров возведения в степень матриц переходных вероятностей для «нелинейной» цепи Маркова представляются ошибочными.

Что касается Марковских шифров второго порядка, то их сходимость к Марковским шифрам первого порядка можно объяснить тем, что для итеративных шифрующих преобразований (шифров) переход к стационарному режиму обозначает статистическую независимость соседних (смежных) значений блоков данных. Поэтому, например, для шифра ГОСТ компонента B_{i-1} в уравнении Марковского процесса второго порядка

$$A_i = f(A_{i-1} \left[+\right] K_j) \oplus B_{i-1}$$

с ростом числа циклов становится случайной по отношению к значениям A_i и A_{i-1}, а это и означает, что уравнение Марковского процесса второго порядка приобретает вид уравнения Марковского процесса первого порядка. Более удивительным,

однако, является тот факт, что шифр сохраняет свойства случайной подстановки и без цикловых подключей (без случайных компонент). Мы на этом моменте остановимся более детально. В табл. 1 представлены результаты экспериментов с шифрами Rijndael, Serpent и ГОСТ по оценке лавинных свойств этих шифров.

В левой части этой таблицы приводятся поцикловые лавинные показатели рассматриваемых шифров (среднее число изменившихся бит на выходе при изменении одиночного бита на входе) при выполнении шифрований с наборами ненулевых значений цикловых подключей, а в правой части таблицы приводятся лавинные показатели этих же шифров с нулевыми цикловыми подключами. Хорошо видно, что результаты как в случае использования ключей, так и в случае их отсутствия практически совпадают. Аналогичные результаты получаются при рассмотрении корреляционных характеристик, а также для дифференциальных и линейных показателей этих же шифров. Вот и получается, что и без случайных компонент шифры всё равно асимптотически ведут себя как случайные подстановки. Но если для Марковских шифров второго порядка есть возможность стать Марковским шифром первого порядка, то, как быть с Марковскими шифрами первого порядка? Получается, что отмеченный результат можно объяснить только тем, что сами цикловые преобразования рассмотренных шифров и без цикловых подключей обладают достаточно эффективным механизмом перемешивания блоков данных. По-видимому, свойство Марковости здесь переходит в сбалансированное статистическое усреднение. Мы возвратимся опять к этому вопросу немного позднее.

Далее будет уместным напомнить новую методологию оценки показателей доказуемой стойкости шифров (Марковских шифров) к атакам дифференциального и линейного криптоанализа, предложенную в нашей работе [4].

3. СУЩНОСТЬ НОВОЙ МЕТОДОЛОГИИ ОЦЕНКИ ДОКАЗУЕМОЙ СТОЙКОСТИ БСШ К АТАКАМ ДИФФЕРЕНЦИАЛЬНОГО И ЛИНЕЙНОГО КРИПТОАНАЛИЗА

В отмеченной работе она сформулирована следующим образом.

Все современные блочные шифры через определенное число циклов независимо от используемых в шифрах S-блоков (*конечно, здесь речь идет не о вырожденных их конструкциях*) *приобретают свойства случайных подстановок, т.е. по комбинаторным показателям* (*числу инверсий, возрастаний и циклов*), *а также по законам распределения переходов таблиц XOR разностей* (*полных дифференциалов*) *и законам распределения смещений таблиц линейных аппроксимаций* (*линейных корпусов*) *повторяют соответствующие показатели случайных подстановок. В результате значения максимумов полных дифференциалов и линейных корпусов могут быть определены расчетным путем из формул для законов распределения вероятностей переходов XOR таблиц и смещений таблиц линейных аппроксимаций случайных подстановок соответствующей степени.*

При этом, проверка показателей случайности больших шифров может быть выполнена на основе разработки и последующего анализа показателей случайности уменьшенных моделей, допускающих проведение вычислительных экспериментов в приемлемые (*реальные*) *сроки.*

Малые модели шифров, повторяющие своих прототипов, позволяют оценить не только средние значения максимумов таблиц дифференциальных вероятностей (*AMDP*), *и средних значений максимумов линейных вероятностей* (*AMLP*) *для ограниченного множества ключей, но и решить задачу определения* (*проверки*) *абсолютного значения максимума по полному множеству ключей.*

Для иллюстрации справедливости приведенных положений мы приведём здесь некоторые примеры анализа дифференциальных и

Таблица 1

Показатели лавинного эффекта для шифров AES, Serpent и ГОСТ при использовании случайных цикловых ключей и безключевые варианты

Число циклов	С ключом			Без ключа		
	AES	Serpent	ГОСТ	AES	Serpent	ГОСТ
1	16,2109	12,1302	2,3666	16,2133	12,137	2,09405
2	64,2589	57,356	5,40601	64,2624	57,3746	4,74361
3	64,0071	63,9972	10,2341	64	63,9977	8,06269
4	64,0062	63,9984	15,5382	64,0033	63,9955	12,4266
5	64,0073	63,9963	21,5132	63,9993	64,0019	17,9388
6	63,9965	64,0038	27,0884	64,0005	63,998	23,2881
7	64,0043	63,9903	30,2263	63,9983	64,0089	27,5203
8	63,9997	63,9951	31,4977	63,9963	64,0028	30,1418
9	64,0038	63,9901	31,8901	63,9979	63,9961	31,3863
10	64,0039	63,9921	31,9901	63,9991	64,0016	31,8744
11	63,9951	64,0013	31,9983	63,9967	64,0057	31,9727
12	63,9937	64,000	32,0002	63,9968	64,0069	31,9948

линейных свойств двух известных шифров AES и ГОСТ 28147-89 и ещё двух шифров Калина и Мухомор, представленных на украинский конкурс (здесь рассматриваются полные версии этих шифров). Длина ключа и блока для AES взята 256 бит, а для шифров Калина и Мухомор длина ключа и блока одинаковые и равны 128 битам.

В табл. 2 представлены поцикловые средние значения максимумов полных дифференциалов для этих шифров, полученные при их использовании в режиме зашифрования 16-битных блоков данных (в качестве результатов зашифрования из зашифрованного блока данных тоже вырезается 16-битный сегмент [5]).

Таблица 2

Поцикловые средние значения максимумов полных дифференциалов шифров вместе со среднеквадратическими отклонениями

Кол-во циклов	AES	ГОСТ 28147-89	Калина	Мухомор
1	65536	65536	65536	19,4
2	3652,26	65536	1382	19,2
3	**19,0666**	61952	**18,8**	19,6
4	19,0666	56008,6	18,8	19,2
5	18,8666	31358	19,2	19
6	19,1332	2046,7	18,8	19,2
7	19,2666	973,4	18,6	19,8
8	19,1332	52,2	18,8	19
9	19,0666	**19,1**	19	19,2
10	19,3333	19,5	18,8	19,6
11	19,4	18,7	19,4	19
13	18,8666	19,1	18,6	18,6
14	18,9332	19,4	19	19,2

В табл. 3 представлены поцикловые средние значения максимумов смещений линейных корпусов этих же шифров вместе со среднеквадратическими отклонениями.

Для всех трех шифров был выполнен подсчет максимумов дифференциалов и смещений линейных корпусов с использованием 10 различных случайно сгенерированных ключей.

В табл. 4 и 5 представлены ещё не опубликованные результаты экспериментов. Здесь рассматривались три шифра: Rijndael, Serpent, Threefish. Опять взяты полные версии данных шифров. Длина ключа и блока для Rijndael и Serpent одинакова и равна 128 битам, а в реализации шифра Threefish использовалась длина для блока и ключа равная 512 битам. Заметим, что в шифре Threefish не применяются S-блоки.

Для всех трех шифров и в этом случае был выполнен подсчет средних значений максимумов полных дифференциалов и смещений линейных корпусов с использованием 10 различных случайно сгенерированных ключей.

Представленные результаты хорошо иллюстрируют переход шифров к стационарному состоянию, повторяющему характеристики случайных подстановок соответствующей степени [11, 12] (наиболее «медленным» оказался шифр Threefish, а вот шифр Мухомор выходит на асимптотические показатели дифференциалов и смещений сразу после первого цикла).

Мы здесь уже не будем приводить результаты экспериментов, свидетельствующие о независимости асимптотических показателей шифров от свойств применяемых в них S-блоков. Отошлём читателей к нашим работам [6,7 и др.]. Остановимся теперь на одном из принципиальных моментов рассматриваемого подхода – приходу шифров к стационарному состоянию свойственному случайной подстановке.

Таблица 3

Поцикловые средние значения максимумов смещений линейных корпусов шифров

Кол-во циклов	AES	ГОСТ 28147-89	Калина	Мухомор
1	0	0	11008,392	**824,742**
2	9284.27	32768	**817,271**	818,621
3	**818.467**	17162	817,718	827,431
4	815	31181,7	814,19	824,193
5	818.5	16150,1	837,349	831,753
6	815.967	16669,5	810,733	814,155
7	832.1	2144,77	820,384	820,975
8	823.133	2380,93	837,917	823,024
9	829.9	**826,833**	809,273	810,196
10	827.4	828,1	821,755	821,316
11	815.6	823,767	827,462	822,385
12	819	821,433	820,291	816,753

Таблица 4

Поцикловые значения максимумов полных дифференциалов для 16-битных сегментов

Число циклов, *r*	MAX (Rijndael)	MAX (Serpent)	MAX (Threefish)
1	16384	18,93	65536
2	8904,25	19,24	65536
3	1911.47	18,64	65536
4	19,24	18,33	42440,04
5	20,31	18,75	30704,23
6	18,83	19,21	9534,57
7	19,21	18,98	37,75
8	19,4	18,37	19,27
9	18,33	19,24	18,78
10	19,17	19,63	18,44

Таблица 5

Математические ожидания максимальных смещений линейных корпусов полных моделей шифров

Число циклов, *r*	MAX (Rijndael)	MAX (Serpent)	MAX (Threefish)
1	0	810,4	32768
2	16313,36	825,0667	32680,93
3	7728,66	828,2667	31306,13
4	817,43	825,9333	23730,93
5	821,98	828,4667	19722,67
6	825,716	824,8667	19722,67
7	817,367	820,3333	7899,8
8	820,167	817,5333	844,0667
9	821,767	820,4	822,1333
10	820,167	816,6	815,8

Выше получен результат (2), в соответствии с которым для Марковского шифра первого порядка формирование матрицы переходных вероятностей всего шифра сводится к последовательному выполнению r однотипных (одноцикловых) преобразований (см. соотношение (3)). По результатам экспериментов можно сделать вывод о том, что *произведение одноцикловых преобразований после небольшого начального числа их повторений приобретает свойства случайной подстановки соответствующей степени независимо от показателей случайности исходного одноциклового преобразования*.

Мы не смогли найти теоретического обоснования этого свойства. Удалось лишь показать [4], что после достижения стационарного распределения при дальнейшем наращивании числа циклов это стационарное распределение сохраняется. Поэтому мы здесь представляем дополнительные обоснования правомерности приведенного утверждения. Наш интерес сосредоточился на процессах, происходящих при последовательном выполнении подстановочных преобразований.

В соответствии с этим далее приводятся результаты вычислительного эксперимента не с шифрами, а с подстановками 256-й степени (байтовыми подстановками). В табл. 6 представлены результаты вычислительного эксперимента по определению максимумов XOR таблиц последовательности подстановочных преобразований для двух байтовых подстановок.

Одна подстановка взята с показателем δ -равномерности равным 4-ём, а вторая с показателем δ -равномерности равным 8-и. Видно, что обе подстановки уже на втором цикле приходят к максимуму дифференциала равному 10-12, характерному для случайной подстановке степени 2^8 [11].

Интересно отметить, что результат не зависит от ключевых значений, если их ввести после каждого подстановочного преобразования.

Конечно, по законам комбинаторики этот процесс должен быть периодическим, но для интересующих нас значений мы, как правило, оказываемся очень далеко от циклового периода подстановки.

Далее мы представляем результат возведения подстановочного преобразования (10 2 0 6 15 1 12 4 14 11 7 13 9 5 3 8) в квадрат (произведения одинаковых полубайтовых подстановочных преобразований:

$$(10\ 2\ 0\ 6\ 15\ 1\ 12\ 4\ 14\ 11\ 7\ 13\ 9\ 5\ 3\ 8)\times$$
$$\times(10\ 2\ 0\ 6\ 15\ 1\ 12\ 4\ 14\ 11\ 7\ 13\ 9\ 5\ 3\ 8)=$$
$$=(7\ 0\ 10\ 12\ 8\ 2\ 9\ 15\ 3\ 13\ 4\ 5\ 11\ 1\ 6\ 14).$$

Ниже представляется произведение матриц переходов (матриц переходных вероятностей умноженных на *AMDP*) этого подстановочного преобразования (возведение матрицы в квадрат).

$$\begin{vmatrix}
16&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&2&2&0&4&0&2&2&2&0&2&0\\
0&0&0&2&2&2&2&0&0&2&4&0&0&2&0&0\\
0&2&2&2&0&0&2&0&0&0&\mathbf{0}&2&4&2&0&0\\
0&0&2&2&2&4&0&2&0&0&0&0&2&0&2&0\\
0&0&2&0&2&0&0&0&0&0&2&4&0&2&2&2\\
0&\mathbf{0}&0&2&0&0&4&2&2&0&0&0&0&2&2&2\\
0&2&2&0&2&0&2&0&2&2&0&0&0&0&4&0\\
0&0&0&0&4&0&2&2&0&2&0&2&2&0&0&2\\
0&4&0&0&2&0&0&2&2&0&2&\mathbf{0}&2&2&0&0\\
0&2&0&0&0&2&0&0&0&2&0&0&2&4&2&2\\
0&0&2&0&0&2&0&4&2&2&0&2&0&2&0&0\\
0&2&\mathbf{0}&4&0&0&0&2&0&2&2&2&0&0&2&0\\
0&2&0&2&2&2&0&0&2&0&0&2&0&0&0&4\\
0&0&2&2&0&0&0&0&2&4&2&0&2&0&0&2\\
0&2&4&0&0&2&2&2&0&0&2&0&\mathbf{0}&0&0&2
\end{vmatrix}\times$$

$$\times\begin{vmatrix}
16&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&2&2&0&4&0&2&2&2&0&2&0\\
0&0&0&2&2&2&2&0&0&2&4&0&0&2&0&0\\
0&2&2&2&0&0&2&0&0&0&\mathbf{0}&2&4&2&0&0\\
0&0&2&2&2&4&0&2&0&0&0&0&2&0&2&0\\
0&0&2&0&2&0&0&0&0&0&2&4&0&2&2&2\\
0&\mathbf{0}&0&2&0&0&4&2&2&0&0&0&0&2&2&2\\
0&2&2&0&2&0&2&0&2&2&0&0&0&0&4&0\\
0&0&0&0&4&0&2&2&0&2&0&2&2&0&0&2\\
0&4&0&0&2&0&0&2&2&0&2&\mathbf{0}&2&2&0&0\\
0&2&0&0&0&2&0&0&0&2&0&0&2&4&2&2\\
0&0&2&0&0&2&0&4&2&2&0&2&0&2&0&0\\
0&2&\mathbf{0}&4&0&0&0&2&0&2&2&2&0&0&2&0\\
0&2&0&2&2&2&0&0&2&0&0&2&0&0&0&4\\
0&0&2&2&0&0&0&0&2&4&2&0&2&0&0&2\\
0&2&4&0&0&2&2&2&0&0&2&0&\mathbf{0}&0&0&2
\end{vmatrix}\neq$$

$$\neq\begin{vmatrix}
16&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&2&0&0&\mathbf{0}&0&4&2&2&0&4&0&0&0&2&0\\
0&2&0&0&0&0&0&2&2&0&0&0&2&6&0&2\\
0&0&0&0&0&2&2&4&0&2&2&4&0&0&0&0\\
0&0&4&4&0&0&0&0&2&0&0&2&2&0&0&\mathbf{2}\\
0&0&2&2&0&6&2&0&2&0&2&0&0&0&0&0\\
0&0&2&2&2&2&0&0&0&0&0&0&0&0&4&4\\
0&0&0&0&2&2&0&\mathbf{0}&4&2&0&2&0&2&2&0\\
0&2&0&4&2&0&0&0&0&2&0&0&0&2&2&2\\
0&0&0&2&0&0&0&2&2&6&2&0&0&0&0&2\\
0&2&2&2&0&2&0&0&0&2&0&0&2&0&4&0\\
0&0&2&0&6&0&2&2&2&0&0&0&0&0&0&2\\
0&2&2&0&0&0&0&0&0&0&2&2&4&2&0&2\\
0&2&0&0&2&2&2&0&0&0&\mathbf{2}&4&2&0&0&0\\
0&4&2&0&0&0&0&2&0&0&2&0&2&2&2&0\\
0&0&0&0&2&0&4&2&0&2&0&2&2&2&0&0
\end{vmatrix}$$

Непосредственное выполнение расчётов показывает, что результат матричного произведения не совпадает с матрицей переходных вероятностей подстановочного преобразования, полученного его умножением самого на себя (она представлена в правой части неравенства).

Таблица 6

Распределение максимумов XOR таблиц
последовательности подстановочных преобразований байтовой подстановки

Число циклов (повторов)	1	2	3	4	5	6	7	8	9	10	11
Значение максимума XOR таблицы для AES S-блока	4	12	12	10	12	12	10	12	12	12	12
Значение максимума XOR таблицы для S-блока Мухомор	8	10	10	12	10	14	12	12	10	12	10

Таким образом, *произведение* (*последовательность*) *подстановочных преобразований нетривиального типа* (*а не только шифров*) *является с большой вероятностью случайной подстановкой, независимо от свойств подстановки, участвующей в формировании этого степенного преобразования.*

Мы посчитали, что это и приведенное выше утверждение является неким «законом природы» (законом хаотического преобразования), который выполняется независимо от нашего желания (может здесь надо более строго оговорить, какие подстановки удовлетворяют этому правилу, но это предмет отдельного исследования).

Аналогично к стационарному распределению свойственному случайной подстановке приходит и любой шифр. Стационарное распределение как раз соответствует тому, что шифр начинает повторять свойства случайной подстановки.

А вот тот факт, что произведение подстановок (и без случайной компоненты), как и последовательность шифрующих преобразований с нулевыми цикловыми подключами, становится случайной подстановкой, оказался всё же неожиданным. Объяснением этому факту может быть лишь то, что сами по себе подстановки (исключая тривиальные их конструкции), как правило, представляют собой набор случайных переходов (уже в самой подстановке заложен механизм случайного перемешивания).

Теперь можно привести дополнительные результаты, уточняющие ряд поднятых ранее вопросов.

О гипотезе статистической эквивалентности.

Как следует из приведенных выше и многочисленных других имеющихся результатов, стационарное состояние, к которому приходит шифр (Марковский шифр) практически не зависит от ключей зашифрования, что хорошо видно из таблиц 1-5. Это говорит о том, что гипотеза статистической эквивалентности, о которой упоминалось в предыдущей нашей работе [1], оказывается справедливой для всех Марковских шифров как первого, так и второго порядков (если их после приведенных особенностей можно называть Марковскими).

Гипотеза статистической эквивалентности в нашей интерпретации выглядит следующим образом:

Гипотеза статистической эквивалентности. *Для Марковских шифров значение максимума полных дифференциалов* (*также как и значение максимума смещений линейных корпусов*) *почти для всех ключей является одним и тем же, практически совпадающим с максимумом таблицы XOR переходов* (*максимумом смещения таблицы линейных аппроксимаций*) *случайной подстановки соответствующей степени.*

Более строгому обоснованию этого положения мы уделим внимание в отдельной работе.

Следующее наше уточнение связано с оценкой показателей стационарности для Марковских шифров.

Возвращаясь к теореме 3 работы [9], мы хотим обратить внимание на то, что по имеющимся многочисленным результатам экспериментов утверждение о том, что с увеличением числа циклов шифр приходит к равномерному стационарному распределению, встречающееся и в ряде других работ, является не верным. На самом деле каждый из известных итеративных шифров после нескольких начальных циклов приходит к стационарным распределениям, повторяющим законы распределения (дифференциалов и смещений таблиц линейных аппроксимаций) случайной подстановки соответствующей степени и дальнейшее наращивание числа циклов шифрующих преобразований не изменяет стационарного распределения [11,12]. В нашей работе [4] приводится теоретическое доказательство этого факта. Поэтому утверждение теоремы 3 во-первых, противоречит вроде бы де признанному многими факту о приходе Марковского шифра к стационарному состоянию, а во-вторых, – в соответствии с приведенными выше и другими имеющимися результатами стационарное состояние шифра соответствует характеристикам случайной подстановки, а для случайной подстановки законы распределения переходов XOR таблиц и смещений таблиц линейных аппроксимаций являются явно не равномерными.

ВЫВОДЫ

1. Как установлено, все итеративные шифры (а они все Марковские) имеют асимптотические значения максимальных значений дифференциалов и линейных корпусов, совпадающие со значениями максимумов XOR таблиц и смещений таблиц линейных аппроксимаций случайных подстановок соответствующей степени.

2. Все Марковские шифры подчиняются закону произведения подстановочных преобразований, в соответствии с которым произведение одноцикловых преобразований после небольшого начального числа их повторений приобретает свойства случайной подстановки соответствующей степени независимо от показателей случайности исходного одноциклового преобразования.

3. Для Марковских шифров асимптотическое значение максимума полных дифференциалов (также как и значение максимума смещений линейных корпусов) почти для всех ключей является одним и тем же, что позволяет при оценке показателей доказуемой стойкости шифров определять асимптотические значения максимумов дифференциальной и линейной вероятности для одного произвольно взятого ключа (или в безключевом варианте).

4. Стационарность распределения переходов таблицы полных дифференциалов также как стационарность смещений линейных корпусов обозначает, что с увеличением числа циклов шифрования эти распределения сохраняются (не меняются).

Литература

[1] *Лисицкая И.В.* Блочные симметричные шифры и Марковские процессы. / И.В. Лисицкая, В.И. Долгов // Прикладная радиоэлектроника, 2012. – Т. 11, № 2. – С. 12-18.

[2] *F. Sano, K. Ohkuma, H. Shimizu, S. Kawamura.* On the Security of Nested SPN Cipher against the Differential and Linear Cryptanalysis/ IEICE Trans. Fundamentals, vol. E86-a, NO.1 January 2003, pp. 37-46.

[3] *Долгов В.И.* Новая методика оценки двухциклового дифференциала уменьшенной версии супер блока AES. / В.И. Долгов, И.В. Лисицкая, В. А. Феськов, К.Е. Лисицкий // Сборник трудов Второй Международной научно-технической конференции «Компьютерные науки и технологии», 8-10 октября 2011, Белгород, С. – 418-422.

[4] *Горбенко І.Д.* Новая идеология оценки стойкости блочных симметричных шифров к атакам дифференциального и линейного криптоанализа. / І.Д. Горбенко, В.И. Долгов, И.В. Лисицкая, Р.В. Олейников // Прикладная радиоэлектроника. – 2010. – Т. 9, № 3. – С. 212-320.

[5] *Лисицкая И.В.* Большие шифры – случайные подстановки / И.В. Лисицкая, А.А. Настенко // Межведомственный научн. технический сборник «Радиотехника». 2011. вып. 166. – С. 50-55.6.

[6] *Кузнецов А.А.* Линейные свойства блочных симметричных шифров, представленных на украинский конкурс. / А.А. Кузнецов, И.В. Лисицкая, С.А. Исаев // Прикладная радиоэлектроника, 2011. – Т. 10, № 2. – С. 135-140.

[7] *Лисицкая И.В.* Об участии S-блоков в формировании максимальных значений дифференциальных вероятностей блочных симметричных шифров. / Лисицкая И.В., Казимиров А.В. // Proceedings International Conference SAIT 2011, Kyiv, Ukraine, May 23-28, 2011, c. 459.

[8] *Лисицкая И.В.* О новой методике оценки стойкости блочных симметричных шифров к атакам дифференциального и линейного криптоанализа. Системы обработки информации. - Харьковский университет Противовоздушных Сил имени Ивана Кожедуба, – 2011. – Вып. 4(94). – С. 167-173.

[9] *X. Lai, J. Massey, and S. Murphy,* Markov ciphers and differentjal cryptanalysis, Advances in Cryptalogy – EUROCRYPT'93, LNCS 547, Springer-Verlag, pp. 17-38, 1991.

[10] *Joan Daemen, Vincent Rijmen* Probability distributions of Correlation and Differentials in Block Ciphers. / Joan Daemen, Vincent Rijmen // April 13, 2006, pp. 1–38.

[11] *Олейников Р.В.* Дифференциальные свойства подстановок / Р.В. Олейников, О.И. Олешко, К.Е. Лисицкий, А.Д. Тевяшев // Прикладная радиоэлектроника. – 2010. – Т.9. – № 3. – С. 326-333.

[12] *Долгов В.И.* Свойства таблиц линейных аппроксимаций случайных подстановок. / В.И. Долгов, И.В. Лисицкая, О.И. Олешко // Прикладная радиоэлектроника. – Харьков: ХНУРЭ. – 2010. – Т. 9 – № 3, С. 334-340.

Поступила в редколлегию 5.03.2012

Лисицкая Ирина Викторовна, фото и сведения об авторе см. на с. 143.

Долгов Виктор Иванович, фото и сведения об авторе см. на с. 143.

Настенко Андрей Александрович, аспирант кафедры БИТ ХНУРЭ. Область научных интересов: криптографическая защита информации.

Лисицкий Константин Евгеньевич, студент 3-го курса кафедры БИТ ХНУРЭ. Область научных интересов: криптографическая защита информации.

УДК 621. 391:519.2:519.7

Оцінки максимальних значень диференціалів та лінійних корпусів Марковських шифрів / В.І. Долгов, І.В. Лисицька, А.А. Настенко, К.Є. Лисицький // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 144–151.

Представляється короткий аналіз сутності відомих підходів до оцінки показників доказової стійкості Марківських блокових симетричних шифрів до атак диференціального та лінійного криптоаналізу. Викладається сутність нової методології оцінки доказової стійкості БСШ до атак диференціального та лінійного криптоаналізу. Наводяться результати оцінки максимальних значень диференціалів та лінійних корпусів Марківських шифрів. Виконується обгрунтування процесу переходу шифрів до стаціонарного стану. Формулюється нова редакція гіпотези статистичної еквівалентності. Уточнюються деякі положення теорії Марківських шифрів, що зустрічаються в публікаціях.

Ключові слова: Марківський шифр, диференціальна ймовірність, лінійна ймовірність, нова методологія оцінки доказової стійкості, стаціонарний стан властивий випадковій підстановці.

Табл. 06. Бібліогр. 12 найм.

UDC 621. 391:519.2:519.7

Estimations of Maximal Values of Differentials and Linear Hulls of Markov Ciphers / V.I. Dolgov, I.V. Lysytska, A.A. Nastenko, K.E. Lysytskiy // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 144–151.

The paper provides a brief analysis of the essence of the known approaches to assessing indicators of provable security of Markov block symmetric ciphers (BSC) to attacks of differential and linear cryptanalysis as well as it describes the essence of the new assessment methodology of BSC demonstrable resistance to the attacks of differential and linear cryptanalysis. The results of evaluating the maximum values of the differentials and linear hulls of Markov ciphers are given. A grounding of the process of BSC transition to the stationary state is performed. A new version of the statistical equivalence hypothesis is formulated. Some points of the theory of Markov ciphers, found in the literature, are particularized.

Keywords: Markov cipher, differential probability, linear probability, new methodology for estimating provable security, steady state characteristic of random substitution.

Tab. 06. Ref. 12 items.

УДК 621.391:519.2:519.7

ОЦЕНКА СЛОЖНОСТИ РАЗЛИЧЕНИЯ СХЕМЫ ЛЕЙ-МЕССИ И СЛУЧАЙНОЙ ПЕРЕСТАНОВКИ

Р.В. ОЛЕЙНИКОВ, Д.С. КАЙДАЛОВ

Выполнен анализ эффективности трехраундовой схемы Лей-Месси – высокоуровневой конструкции симметричного блочного шифра. Оценка получена на основе определения сложности проведения атаки с выбранными открытыми текстами, направленной на различение конструкции криптографического преобразования и модели идеального шифра – случайной перестановки. Обоснована верхняя граница преимущества для произвольного алгоритма-различителя и точные значения преимущества для двух конкретных методов.

Ключевые слова: блочный шифр, схема Лей-Месси, случайная перестановка.

ВВЕДЕНИЕ

Схема Лей-Месси является альтернативной высокоуровневой конструкцией блочных симметричных шифров. На ее основе построены алгоритмы FOX, «Мухомор» и др. Основным преимуществом схемы Лей-Месси, как и цепи Фейстеля, является возможность построения инволютивного преобразования, т.е. расшифрование реализовано практически аналогично зашифрованию при использовании обратного порядка раундовых подключей. Дополнительным преимуществом этой конструкции является отсутствие требований к биективности раундовой функции (как у SPN-структур), что упрощает разработку и реализацию.

Как и цепь Фейстеля, схема Лей-Месси представляет собой итеративную структуру. Операции шифрования можно выразить следующим образом [1]:

$$L_i = \sigma(L_{i-1} \oplus F(L_{i-1} \oplus R_{i-1}, K_{i-1})),$$

$$R_i = R_{i-1} \oplus F(L_{i-1} \oplus R_{i-1}, K_{i-1}), \text{ при } i \in 1...n-1.$$

На последнем раунде преобразование σ отсутствует:

$$L_n = L_{n-1} \oplus F(L_{n-1} \oplus R_{n-1}, K_{n-1}),$$

$$R_n = R_{n-1} \oplus F(L_{n-1} \oplus R_{n-1}, K_{n-1}).$$

Здесь R_i и L_i – это правая и левая части сообщения на i-ом раунде (соответственно R_0 и L_0 – открытое сообщение, а R_n и L_n – зашифрованное сообщение, где n – количество раундов). σ –некоторое ортоморфное преобразование [1], а отображение F – это функция усложнения, зависимая от ключа. Общая схема одного раунда приведена на рис. 1.

Далее предполагается, что σ построена на основе одного раунда цепи Фейстеля с тождественной раундовой функцией (рис. 2), поскольку в шифрах, построенных по схеме Лей-Месси (FOX, Мухомор), используется именно такая конструкция.

Блочный шифр реализует определенное подмножество перестановок, количество которых равняется количеству возможных ключей шифрования. Т.к. выбор такого подмножества определяется структурой шифра и не является случайным, то возможно построение алгоритма-различителя, который мог бы определить, является ли конкретная перестановка случайно выбранной из общего множества, либо полученной в результате действия блочного шифра. Таким образом, анализируя входные данные, поданные на вход алгоритма-различителя, на выходе алгоритма мы будем иметь «1» либо «0». «1» в том случае, если считается, что входные данные были получены с помощью блочного шифра (схемы Лей-Месси в данном случае) и «0», если считается, что это результат действия случайной функции. Для схемы Лей-Месси вероятность появления «1» будет иметь определенное значение. Однако и для случайной функции вероятность появления «1» на выходе алгоритма-различителя не равна нулю, поскольку возможен случайный выбор произвольной перестановки, аналогичной сформированной схемой Лей-Месси.

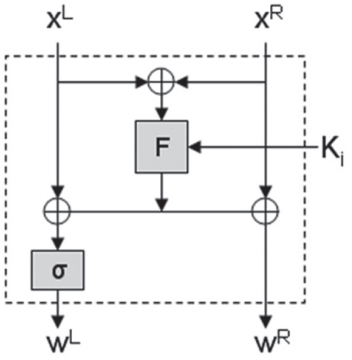


Рис. 1. Схема Лей-Месси

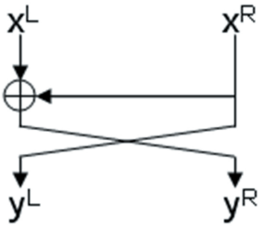


Рис. 2. Функция σ

Детально модель алгоритма-различителя и характеристики его эффективности рассмотрены в нашей предыдущей статье [2].

1. ВЕРХНЯЯ ГРАНИЦА ВЕРОЯТНОСТИ РАЗЛИЧЕНИЯ СХЕМЫ ЛЕЙ-МЕССИ И СЛУЧАЙНОЙ ФУНКЦИИ

Далее будет рассмотрена максимальная вероятность различения случайной функции и схемы Лей-Месси с количеством раундов $r \geq 3$. Доказательство будет приводиться для 3-раундовой схемы Лей-Месси, однако итоговое неравенство будет справедливо и для конструкций с большим количеством раундов, поскольку сложность различения будет только увеличиваться.

3-раундовая схема Лей-Месси приведена на рис. 3.

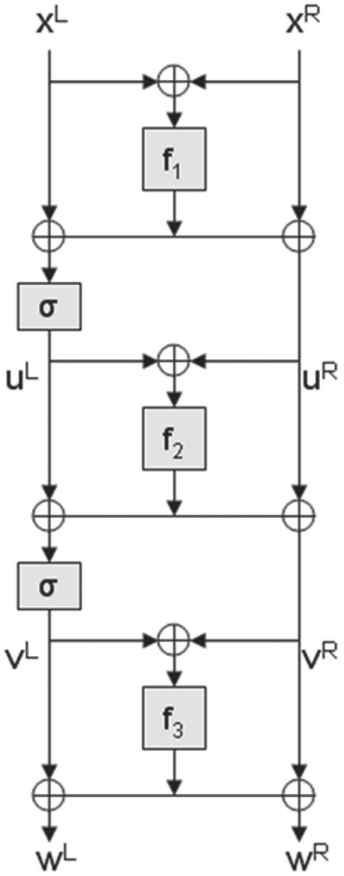


Рис. 3. Схема Лей-Месси

Далее используются следующие обозначения:

x^L, x^R – левая и правая половины входного блока данных, каждая по n бит. Соответственно общая длина входного блока составляет $2n$ бит;

$\Delta x = x^L \oplus x^R$ – XOR-разность между левой и правой половинами битового вектора;

u^L, u^R – левая и правая половины промежуточного результата после 1-го раунда преобразований;

v^L, v^R – левая и правая половины промежуточного результата после 2-го раунда преобразований;

w^L, w^R – левая и правая половины выходного значения;

f_1, f_2, f_3 – случайные функции;

$\sigma^2(x) = \sigma(\sigma(x))$.

Кроме того, вводится функция $\sigma'(x) = \sigma(x) + x$. Если σ – линейная, то σ' тоже линейная.

Отметим, что рассматриваемый алгоритм-различитель будет иметь смысл для любой линейной функции (т.е. такой, что $\sigma(x+y) = \sigma(x) + \sigma(y)$).

Теорема 1. Максимальная вероятность различения схемы Лей-Месси (рис. 3) с количеством раундов $r \geq 3$ и случайных раундовых функциях при k запросах не превышает значения

$$Adv_{\max}(LM, PRF) \leq 1 - \left(\frac{2^n - 2}{2^n - 1}\right)^{\frac{k(k-1)}{2}}.$$

Доказательство.

Сначала необходимо доказать, что при $\Delta U_i \neq \Delta U_j$ (рис. 1) различение невозможно.

Итак, пусть $\Delta U_i \neq \Delta U_j$ для всех возможных пар запросов. Тогда

$$W_i^L = V_i^L \oplus f_3(\Delta V_i) =$$
$$= \sigma^2(x_i^L \oplus f_1(\Delta x_i)) \oplus \sigma(f_2(\Delta U_i)) \oplus f_3(\Delta V_i),$$
$$W_i^R = V_i^R \oplus f_3(\Delta V_i) =$$
$$= x_i^R \oplus f_1(\Delta x_i) \oplus f_2(\Delta U_i) \oplus f_3(\Delta V_i),$$
$$W_j^L = V_j^L \oplus f_3(\Delta V_j) =$$
$$= \sigma^2(x_j^L \oplus f_1(\Delta x_j)) \oplus \sigma(f_2(\Delta U_j)) \oplus f_3(\Delta V_j),$$
$$W_j^R = V_j^R \oplus f_3(\Delta V_j) =$$
$$= x_j^R \oplus f_1(\Delta x_j) \oplus f_2(\Delta U_j) \oplus f_3(\Delta V_j).$$

Поскольку f_2 – случайная функция (перестановка), то $f_2(\Delta U_i)$ и $f_2(\Delta U_j)$ – случайные величины. Следовательно, выходные значения схемы Лей-Месси также будут случайными.

Кроме того, рассмотрим значения ΔW_i и ΔW_i:

$$\Delta W_i = W_i^L \oplus W_i^R = \sigma^2(x_i^L \oplus f_1(\Delta x_i)) \oplus \sigma(f_2(\Delta U_i)) \oplus$$
$$\oplus x_i^R \oplus f_1(\Delta x_i) \oplus f_2(\Delta U_i) = \sigma^2(x_i^L \oplus f_1(\Delta x_i)) \oplus$$
$$\oplus x_i^R \oplus f_1(\Delta x_i) \oplus \sigma'(f_2(\Delta U_i)),$$

$$\Delta W_j = W_j^L \oplus W_j^R = \sigma^2(x_j^L \oplus f_1(\Delta x_j)) \oplus$$
$$\oplus \sigma(f_2(\Delta U_j)) \oplus x_j^R \oplus f_1(\Delta x_j) \oplus f_2(\Delta U_j) =$$
$$= \sigma^2(x_j^L \oplus f_1(\Delta x_j)) \oplus x_j^R \oplus f_1(\Delta x_j) \oplus \sigma'(f_2(\Delta U_j)).$$

Поскольку f_2 – случайная функция (перестановка), то $\sigma'(f_2(\Delta U_i))$ и $\sigma'(f_2(\Delta U_j))$ – случайные величины. Соответственно ΔW_i и ΔW_i в таком случае тоже случайные значения.

Вероятность выполнения условия $\Delta U_i = \Delta U_j$ можно оценить следующим образом.

$$\sigma(x_i^L \oplus f_1(\Delta x_i)) \oplus x_i^R \oplus f_1(\Delta x_i) =$$
$$= \sigma(x_j^L \oplus f_1(\Delta x_j)) \oplus x_j^R \oplus f_1(\Delta x_j),$$

$$\sigma(x_i^L) \oplus \sigma(f_1(\Delta x_i)) \oplus x_i^R \oplus f_1(\Delta x_i) =$$
$$= \sigma(x_j^L) \oplus \sigma(f_1(\Delta x_j)) \oplus x_j^R \oplus f_1(\Delta x_j),$$

$$\begin{aligned}\sigma(f_1(\Delta x_i))\oplus\sigma(f_1(\Delta x_j))\oplus f_1(\Delta x_i)\oplus f_1(\Delta x_j)=\\ =\sigma(x_j^L)\oplus\sigma(x_i^L)\oplus x_j^R\oplus x_i^R,\\ \sigma^{/}(f_1(\Delta x_i))\oplus\sigma(f_1(\Delta x_j))=\\ =\sigma(x_j^L)\oplus\sigma(x_i^L)\oplus x_j^R\oplus x_i^R.\end{aligned} \quad (1)$$

Если f_1 – случайная функция, то максимальная вероятность выполнения (1) (при подобранных открытых текстах) не превышает

$$P_1 \le \frac{1}{2^n}.$$

Если же f_1 – случайная перестановка, то максимальная вероятность выполнения (1) (при подобранных открытых текстах) не превышает

$$P_2 \le \frac{1}{2^n-1}. \quad (2)$$

Формула (2) учитывает условие

$$\sigma^{/}(f_1(\Delta x_i))\oplus\sigma(f_1(\Delta x_j))\neq 0.$$

Поскольку определяется верхняя граница различения, то $P(\Delta U_i=\Delta U_j)=P_2$. Тогда для двух запросов вероятность выполнения условия $\Delta U_i\neq\Delta U_j$ равна

$$P(\Delta U_i\neq\Delta U_j)=1-P_2\le 1-\frac{1}{2^n-1}=\frac{2^n-2}{2^n-1}.$$

Для k запросов вероятность того, что для каждой пары выполняется условие $\Delta U_i\neq\Delta U_j$, имеет следующее значение:

$$P(\Delta U_i\neq\Delta U_j)\le\left(\frac{2^n-2}{2^n-1}\right)^{\frac{k(k-1)}{2}}. \quad (3)$$

Отсюда вероятность различения схемы Лей-Месси не превышает значения (3).

Максимальное преимущество алгоритма-различителя равно

$$Adv_{\max}(LM,PRF)=|P_{\max}(LM)-P_{\max}(PRF)|.$$

Можно сделать предположение в пользу криптоаналитика о том, что для случайной функции вероятность появления «1» на выходе алгоритма-различителя равна нулю (ложное срабатывание не допускается). Тогда преимущество равно

$$\begin{aligned}Adv_{\max}(LM,PRF)=|P_{\max}(LM)-P_{\max}(PRF)\le|\\ \le 1-\left(\frac{2^n-2}{2^n-1}\right)^{\frac{k(k-1)}{2}}.\end{aligned} \quad (4)$$

Доказательство окончено.

На рис. 4 и 5 приведены теоретически максимальные вероятности различения цепи Фейстеля (пунктиром) и схемы Лей-Месси. Формула для максимальной вероятности различения цепи Фейстеля рассматривалась в [2]. Для схемы Лей-Месси (сплошная линия) использовалась формула (4).

Из рис. 4 и 5 видно, что при одинаковом количестве запросов вероятность различения схемы Лей-Месси ниже, что свидетельствует о более высокой эффективности схемы Лей-Месси как высокоуровневой конструкции блочного шифра.

Рассмотренная вероятность является всего лишь теоретической верхней границей различения. Это значит, что реальная максимальная вероятность различения может быть и меньше.

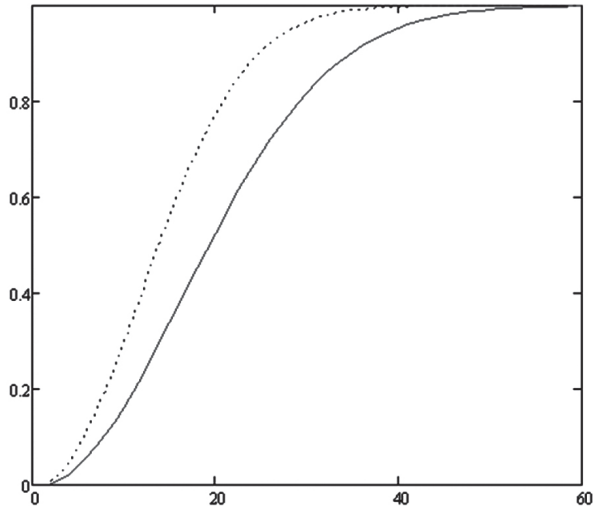


Рис. 4. Максимальные вероятности различения для цепи Фейстеля и схемы Лей-Месси для блока n=8

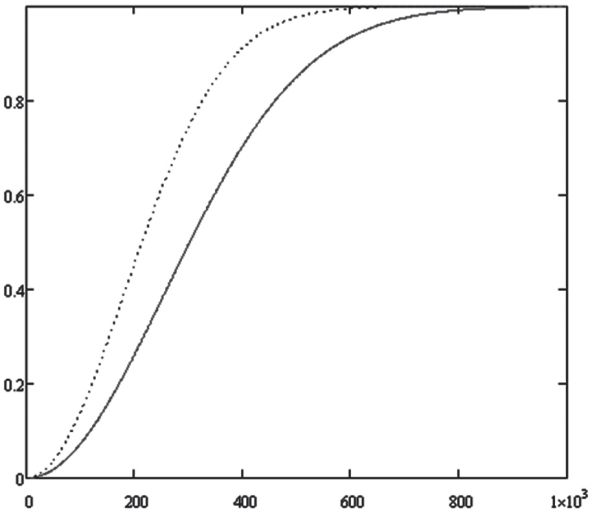


Рис. 5. Максимальные вероятности различения для цепи Фейстеля и схемы Лей-Месси для блока n=16

2. АЛГОРИТМ-РАЗЛИЧИТЕЛЬ ДЛЯ 3-РАУНДОВОЙ СХЕМЫ ЛЕЙ-МЕССИ СО СЛУЧАЙНЫМИ ФУНКЦИЯМИ В КАЧЕСТВЕ РАУНДОВЫХ ПРЕОБРАЗОВАНИЙ

Оценка вероятности различения, приведенная в теореме 1, является теоретически максимально возможной вероятностью различения. Однако для практически реализуемых алгоритмов различения данная вероятность будет меньше. Кроме того, с ростом числа раундов вероятность различения также будет уменьшаться. Стоит отметить, что высокоуровневая конструкция может иметь несколько различителей одновременно, каждый из которых позволяет получить определенную вероятность.

Алгоритм-различитель №1 для 3-раундовой схемы Лей-Месси со случайными функциями в качестве раундовых преобразований.

Алгоритм-различитель для k пар входных аргументов (x_i, x_j) проверяет выполнение равенства

$$\Delta W_i \oplus \Delta W_j = \sigma(x_i^R \oplus x_j^R) \oplus x_i^R \oplus x_j^R . \qquad (5)$$

В случае выполнения хотя бы для одной пары возвращаемое значение будет «1» (определена схема Лей-Месси), иначе – «0» (случайная перестановка).

Теорема 2. Для k запросов вида (x_i, x_j), соответствующих условиям $\sigma(x_i^L \oplus x_j^L) = x_i^R \oplus x_j^R$ и $\Delta x_i \neq \Delta x_j$, при проверке выполнения равенства $\Delta W_i \oplus \Delta W_j = \sigma(x_i^R \oplus x_j^R) \oplus x_i^R \oplus x_j^R$, $0 \le i < j < k$ вероятность различения 3-раундовой схемы Лей-Месси на основе случайных функций как раундового преобразования и случайной перестановки не превышает значения

$$Adv(LM, PRP) \le$$

$$\le \left| \left(1 - \frac{2^n}{2^{2n}-1}\right)^{\frac{k(k-1)}{2}} - \left(1 - \frac{2^{n+1}-1}{2^{2n}}\right)^{\frac{k(k-1)}{2}} \right|. \qquad (6)$$

Доказательство.

При $k = 2$ для схемы Лей-Месси вероятность выполнения равенства (5) при использовании случайных функций в раундовом преобразовании равна значению

$$P(LM) = 1 - \left(1 - \frac{1}{2^n}\right)^2 = \frac{2^{n+1}-1}{2^{2n}} .$$

Результат получается из следующих выводов. Полные выражения для ΔW_i и ΔW_j представляются в виде

$$\Delta W_i = W_i^L \oplus W_i^R = V_i^L \oplus V_i^R =$$
$$= \sigma(\sigma(x_i^L \oplus f_1(\Delta x_i)) \oplus f_2(\Delta U_i)) \oplus \qquad (7)$$
$$\oplus x_i^R \oplus f_1(\Delta x_i) \oplus f_2(\Delta U_i),$$

$$\Delta W_j = W_j^L \oplus W_j^R = V_j^L \oplus V_j^R =$$
$$= \sigma(\sigma(x_j^L \oplus f_1(\Delta x_j)) \oplus f_2(\Delta U_j)) \oplus \qquad (8)$$
$$\oplus x_j^R \oplus f_1(\Delta x_j) \oplus f_2(\Delta U_j).$$

Учитывая, что функция - линейная, выражения (7) и (8) можно представить следующим образом:

$$\Delta W_i = W_i^L \oplus W_i^R = V_i^L \oplus V_i^R =$$
$$= \sigma^2(x_i^L) \oplus \sigma^2(f_1(\Delta x_i)) \oplus \qquad (9)$$
$$\oplus \sigma(f_2(\Delta U_i)) \oplus x_i^R \oplus f_1(\Delta x_i) \oplus f_2(\Delta U_i),$$

$$\Delta W_j = W_j^L \oplus W_j^R = V_j^L \oplus V_j^R =$$
$$= \sigma^2(x_j^L) \oplus \sigma^2(f_1(\Delta x_j)) \oplus \sigma(f_2(\Delta U_j)) \oplus \qquad (10)$$
$$\oplus x_j^R \oplus f_1(\Delta x_j) \oplus f_2(\Delta U_j).$$

Таким образом, равенство (5) будет выполняться в двух случаях:

1. В случае возникновения коллизии на функции f_1: если произошла коллизия и $f_1(\Delta x_i) = f_1(\Delta x_j)$, то также выполняется условие $\Delta U_i = \Delta U_j$, соответственно и $f_2(\Delta U_i) = f_2(\Delta U_j)$.

Для равенства $\Delta U_i = \Delta U_j$:

$$\Delta U_i = U_i^L \oplus U_i^R =$$
$$= \sigma(x_i^L \oplus f_1(\Delta x_i)) \oplus x_i^R \oplus f_1(\Delta x_i),$$
$$\Delta U_j = U_j^L \oplus U_j^R =$$
$$= \sigma(x_j^L \oplus f_1(\Delta x_j)) \oplus x_j^R \oplus f_1(\Delta x_j).$$

Соответственно, если $\Delta U_i = \Delta U_j$, то справедливо следующее выражение:

$$\sigma(x_i^L) \oplus \sigma(f_1(\Delta x_i)) \oplus x_i^R \oplus f_1(\Delta x_i) =$$
$$= \sigma(x_j^L) \oplus \sigma(f_1(\Delta x_j)) \oplus x_j^R \oplus f_1(\Delta x_j).$$

Из $f_1(\Delta x_i) = f_1(\Delta x_j)$ следует

$$\sigma(x_i^L) \oplus x_i^R = \sigma(x_j^L) \oplus x_j^R .$$

Это соответствует условию, по которому отбираются пары открытых текстов, поэтому при возникновении коллизии на f_1 также выполняется $f_2(\Delta U_i) = f_2(\Delta U_j)$.

В этом случае разность выражений (9) и (10) равна

$$\Delta W_i \oplus \Delta W_j = \sigma^2(x_i^L) \oplus x_i^R \oplus \sigma^2(x_j^L) \oplus x_j^R =$$
$$= \sigma^2(x_i^L \oplus x_j^L) \oplus x_i^R \oplus x_j^R ,$$

что соответствует (5). Это означает, что при коллизии на функции f_1 выражение (5) действительно выполняется.

f_1 – случайная функция и $\Delta x_i \neq \Delta x_j$ по условию, поэтому вероятность коллизии равна

$$P(f_1(\Delta x_i) = f_1(\Delta x_j)) = \frac{1}{2^n} .$$

2. В случае отсутствия коллизии на функции f_1 и выполнения следующего равенства:

$$\sigma^2(f_1(\Delta x_i) \oplus f_1(\Delta x_j)) \oplus f_1(\Delta x_i) \oplus f_1(\Delta x_j) =$$
$$= \sigma'(f_2(\Delta U_i) \oplus f_2(\Delta U_j)).$$

Полное уравнение для $\Delta W_i \oplus \Delta W_j$ имеет вид

$$\Delta W_i \oplus \Delta W_j = \sigma^2(x_i^L) \oplus \sigma^2(f_1(\Delta x_i)) \oplus$$
$$\oplus \sigma(f_2(\Delta U_i)) \oplus x_i^R \oplus f_1(\Delta x_i) \oplus f_2(\Delta U_i) \oplus$$
$$\oplus \sigma^2(x_j^L) \oplus \sigma^2(f_1(\Delta x_j)) \oplus \sigma(f_2(\Delta U_j)) \oplus$$
$$\oplus x_j^R \oplus f_1(\Delta x_j) \oplus f_2(\Delta U_j) =$$
$$= \sigma^2(x_i^L \oplus x_j^L) \oplus x_i^R \oplus x_j^R \oplus \sigma^2(f_1(\Delta x_i) \oplus$$
$$\oplus f_1(\Delta x_j)) \oplus \sigma(f_2(\Delta U_i) \oplus f_2(\Delta U_j)) \oplus$$
$$\oplus f_1(\Delta x_i) \oplus f_1(\Delta x_j) \oplus f_2(\Delta U_i) \oplus f_2(\Delta U_j).$$

Отсюда следует

$$\Delta W_i \oplus \Delta W_j \oplus \sigma^2(x_i^L \oplus x_j^L) \oplus x_i^R \oplus x_j^R =$$
$$= \sigma^2(f_1(\Delta x_i) \oplus f_1(\Delta x_j)) \oplus f_1(\Delta x_i) \oplus$$
$$\oplus f_1(\Delta x_j) \oplus \sigma'(f_2(\Delta U_i) \oplus f_2(\Delta U_j)).$$

Таким образом, для выполнения условия (5) должно выполняться равенство:

$$\sigma^2(f_1(\Delta x_i) \oplus f_1(\Delta x_j)) \oplus f_1(\Delta x_i) \oplus$$
$$\oplus f_1(\Delta x_j) \oplus \sigma'(f_2(\Delta U_i) \oplus f_2(\Delta U_j)) = 0,$$

$$\sigma^2(f_1(\Delta x_i)\oplus f_1(\Delta x_j))\oplus f_1(\Delta x_i)\oplus f_1(\Delta x_j)= \\ =\sigma'(f_2(\Delta U_i)\oplus f_2(\Delta U_j)). \quad (11)$$

Поскольку функции f_1 и f_2 – случайные, то вероятность выполнения уравнения (11) имеет значение

$$P=\frac{1}{2^n}.$$

Исходя из двух описанных случаев, можно найти общую вероятность выполнения равенства (5) для схемы Лей-Месси при одной паре входных запросов:

$$P_1(LM)=1-(1-\frac{1}{2^n})^2=\frac{2^{n+1}-1}{2^{2n}}.$$

Если принять, что вероятность выполнения равенства (5) для всех пар запросов одинаковая, то общую вероятность получения «1» на выходе алгоритма-различителя можно получить с помощью следующего выражения:

$$P_2(LM)=1-(1-\frac{2^{n+1}-1}{2^{2n}})^{\frac{k(k-1)}{2}}, \quad (12)$$

где k – количество запросов. Вероятность определяется по формуле для независимых событий, т.к. одновременно несколько независимых пар могут удовлетворять заданному равенству. Общее количество возможных пар составляет $C_k^2=\frac{k(k-1)}{2}$.

Формула (12) является аппроксимационным значением, имеющим минимальную погрешность при количестве запросов, значительно меньшим мощности множества открытых (шифрованных) текстов.

Точное значение вероятности можно получить с помощью формулы

$$P_3(LM)=1-\prod_{i=0}^{k-2}(1-\frac{1}{2^n-i})^{2(k-(i+1))}. \quad (13)$$

Это связано с тем, что пары запросов не являются независимыми и равновероятными. Однако отличия в вероятностях минимальны, поэтому для упрощения целесообразно пользоваться аппроксимированным значением (12). Зависимости между запросами рассматривались в нашей предыдущей работе [2].

Т.к. аппроксимационная формула дает большее значение вероятности различения, чем точная формула, то верхнюю границу целесообразно задать как неравенство, что и сделано в формуле (6). В дальнейшем для упрощения формул будет использоваться именно аппроксимационное значение вероятности.

Для случайной перестановки вероятность выполнения равенства (5) при $k=2$ запросах равна

$$P(PRP)=\frac{2^n}{2^{2n}-1}.$$

Вычитание единицы из знаменателя обусловлено тем, что в перестановке при разных входных значениях не повторяются выходные значения. Т.е. $W_i\neq W_j$, и соответственно, невозможен случай, когда $W_i^L\oplus W_j^L=W_i^R\oplus W_j^R=0$. Поскольку правая часть выражения (5) никогда не равна нулю (по условию для линейного ортоморфного преообразования), то вероятность увеличивается.

Если принять, что вероятность выполнения равенства (5) для всех пар запросов одинаковая, то общую вероятность получения «1» на выходе алгоритма-различителя для случайной перестановки можно получить с помощью выражения:

$$P_2(LM)=1-(1-\frac{2^n}{2^{2n}-1})^{\frac{k(k-1)}{2}}.$$

Способ нахождения такой вероятности аналогичен описанному выше. Точное значение вероятности можно получить, используя следующее выражение:

$$P_3(LM)=1-\prod_{i=0}^{k-2}(1-\frac{2^n}{2^{2n}-1-i\cdot 2^n})^{k-(i+1)}. \quad (14)$$

Доказательство этой формулы уже приводилось в нашей предыдущей статье [2] и здесь опускается.

Преимущество алгоритма-различителя находится как модуль разности значений (13) и (14):

$$Adv(LM,PRP)=|\prod_{i=0}^{k-2}(1-\frac{1}{2^n-i})^{2(k-(i+1))}- \\ -\prod_{i=0}^{k-2}(1-\frac{2^n}{2^{2n}-1-i\cdot 2^n})^{k-(i+1)}|\leq \\ \leq|(1-\frac{2^n}{2^{2n}-1})^{\frac{k(k-1)}{2}}-(1-\frac{2^{n+1}-1}{2^{2n}})^{\frac{k(k-1)}{2}}|.$$

Доказательство окончено.

Следует отметить, что увеличение количества запросов до определенного порогового значения приводит к возрастанию вероятности различения. Дальнейшие запросы снижают эффективность различения.

На рис. 6 и 7 приведен график зависимости вероятности (ось ординат) различения для $n=8$ и $n=16$ от количества входных запросов (ось абсцисс).

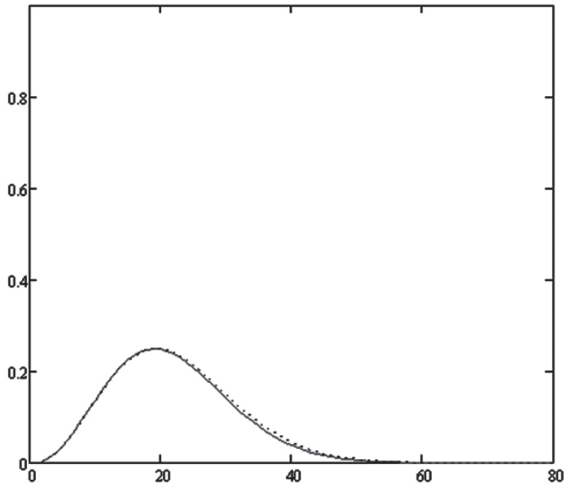


Рис. 6. Зависимость вероятности различения от количества входных запросов для $n=8$

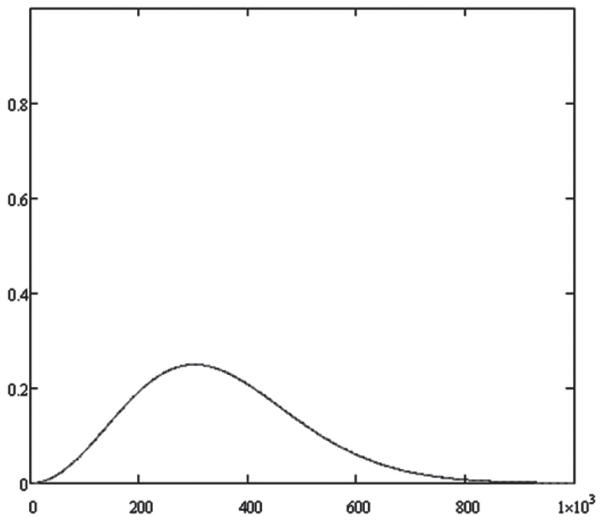


Рис. 7. Зависимость вероятности различения от количества входных запросов для $n = 16$

Как следует из графиков, для выбранных параметров с помощью описанного алгоритма-различителя можно достичь вероятности различения порядка $P_{\max} \approx 0.3$. При этом требуется порядка $\sqrt{2^n}$ выбранных открытых текстов.

3. АЛГОРИТМ-РАЗЛИЧИТЕЛЬ ДЛЯ 3-РАУНДОВОЙ СХЕМЫ ЛЕЙ-МЕССИ СО СЛУЧАЙНЫМИ ПЕРЕСТАНОВКАМИ В КАЧЕСТВЕ РАУНДОВЫХ ПРЕОБРАЗОВАНИЙ

Используется та же схема (рис. 3), только раундовые функции f_1, f_2, f_3 – случайные перестановки. В этом случае более эффективно использовать следующий алгоритм-различитель.

Алгоритм-различитель №2 для 3-раундовой схемы Лей-Месси со случайными перестановками в качестве раундовых преобразований.

Алгоритм-различитель для k пар входных аргументов (x_i, x_j) проверяет выполнение равенства

$$\Delta W_i \oplus \Delta W_j = \sigma(x_i^R \oplus x_j^R \oplus x_i^L \oplus x_j^L),$$
$$0 \le i < j < k. \qquad (15)$$

Если хотя бы для одной пары данное равенство выполняется, то на выходе алгоритм выдает «1» (определена цепь Лей-Месси), иначе – «0» (случайная перестановка).

Теорема 3. Для k запросов вида (x_i, x_j), соответствующих условиям

$$\sigma(x_i^L \oplus x_j^L) \oplus x_i^R \oplus x_j^R \neq 0 \text{ и } \Delta x_i \neq \Delta x_j,$$

при проверке выполнения равенства

$$\Delta W_i \oplus \Delta W_j = \sigma(x_i^R \oplus x_j^R \oplus x_i^L \oplus x_j^L),$$

$0 \le i < j < k$ вероятность различения 3-раундовой схемы Лей-Месси на основе случайных перестановок как раундового преобразования и случайной перестановки не превышает значения

$$Adv(LM, PRP) \le \left|(1 - \frac{2^n}{2^{2n}-1})^{\frac{k(k-1)}{2}} - (1 - \frac{1}{2^n - 1})^{k(k-1)}\right|.$$

Доказательство.

Вероятность выполнения условия

$$\Delta W_i \oplus \Delta W_j = \sigma(x_i^R \oplus x_j^R \oplus x_i^L \oplus x_j^L)$$

для схемы Лей-Месси при $k = 2$ равна

$$P(LM) = 1 - (1 - \frac{1}{2^n - 1})^2.$$

Результат получается из следующих выводов. Равенство (15) выполняется в двух случаях:

1. В случае возникновения коллизии $\Delta U_i = \Delta U_j$. Тогда справедливо следующее:

$$\begin{gathered}\sigma(x_i^L \oplus f_1(\Delta x_i)) \oplus x_i^R \oplus f_1(\Delta x_i) = \\ = \sigma(x_j^L \oplus f_1(\Delta x_j)) \oplus x_j^R \oplus f_1(\Delta x_j), \\ \sigma(x_i^L) \oplus \sigma(f_1(\Delta x_i)) \oplus x_i^R \oplus f_1(\Delta x_i) = \\ = \sigma(x_j^L) \oplus \sigma(f_1(\Delta x_j)) \oplus x_j^R \oplus f_1(\Delta x_j), \\ \sigma(f_1(\Delta x_i)) \oplus \sigma(f_1(\Delta x_j)) \oplus f_1(\Delta x_i) \oplus f_1(\Delta x_j) = \\ = \sigma(x_j^L) \oplus \sigma(x_i^L) \oplus x_j^R \oplus x_i^R. \end{gathered} \qquad (16)$$

Поскольку $\Delta x_i \neq \Delta x_j$, а f_1, f_2, f_3 – это перестановки, то для всех запросов

$$\sigma(f_1(\Delta x_i)) \oplus \sigma(f_1(\Delta x_j)) \oplus f_1(\Delta x_i) \oplus f_1(\Delta x_j) \neq 0. \qquad (17)$$

Из (16) и (17) также следует, что $\sigma(x_j^L) \oplus \sigma(x_i^L) \oplus x_j^R \oplus x_i^R \neq 0$ – начальное условие для входных текстов.

Если произошла коллизия и равенство (16) выполняется, то

$$\begin{gathered}\Delta W_i \oplus \Delta W_j = \sigma^2(x_i^L) \oplus \sigma^2(f_1(\Delta x_i)) \oplus \\ \oplus \sigma(f_2(\Delta U_i)) \oplus x_i^R \oplus f_1(\Delta x_i) \oplus f_2(\Delta U_i) \oplus \\ \oplus \sigma^2(x_j^L) \oplus \sigma^2(f_1(\Delta x_j)) \oplus \sigma(f_2(\Delta U_j)) \oplus \\ \oplus x_j^R \oplus f_1(\Delta x_j) \oplus f_2(\Delta U_j) = \\ = \sigma^2(x_i^L \oplus x_j^L) \oplus x_i^R \oplus x_j^R \oplus \sigma^2(f_1(\Delta x_i) \oplus \\ \oplus f_1(\Delta x_j)) \oplus f_1(\Delta x_i) \oplus f_1(\Delta x_j). \end{gathered} \qquad (18)$$

Из (16) и (18) следует

$$\Delta W_i \oplus \Delta W_j = \sigma(x_i^L \oplus x_j^L \oplus x_i^R \oplus x_j^R). \qquad (19)$$

Равенство (19) соответствует условию различения, т.е. оно справедливо при выполнении (16). Вероятность выполнения (16) равна

$$P_1(LM) = \frac{1}{2^n - 1}.$$

Уменьшение знаменателя на единицу обусловлена $\sigma(x_j^L) \oplus \sigma(x_i^L) \oplus x_j^R \oplus x_i^R \neq 0$ для входных текстов. Это увеличивает вероятность, поскольку левая часть (16) никогда не равна нулю.

2. В случае отсутствия коллизии $\Delta U_i = \Delta U_j$ и при выполнении следующего равенства:

$$\begin{gathered}\Delta W_i \oplus \Delta W_j = \sigma^2(x_i^L) \oplus \sigma^2(f_1(\Delta x_i)) \oplus \\ \oplus \sigma(f_2(\Delta U_i)) \oplus x_i^R \oplus f_1(\Delta x_i) \oplus f_2(\Delta U_i) \oplus \\ \oplus \sigma^2(x_j^L) \oplus \sigma^2(f_1(\Delta x_j)) \oplus \sigma(f_2(\Delta U_j)) \oplus \\ \oplus x_j^R \oplus f_1(\Delta x_j) \oplus f_2(\Delta U_j) = \\ = \sigma(x_i^L \oplus x_i^R \oplus x_j^L \oplus x_j^R). \end{gathered} \qquad (20)$$

Поскольку функции f_1 и f_2 – это случайные перестановки, а $\Delta W_i \oplus \Delta W_j$ – случайное значение (с некоторыми оговорками), то вероятность выполнения уравнения (20) имеет следующее значение:

$$P = \frac{2^n + 1}{2^{2n} - 1} = \frac{1}{2^n - 1}.$$

Из двух описанных условий можно найти общую вероятность выполнения равенства (15) для схемы Лей-Месси со случайными раундовыми перестановками при одной паре входных запросов. Сумма вероятностей находится по формуле для независимых событий:

$$P_1(LM) = 1 - (1 - \frac{1}{2^n - 1})^2.$$

Если принять, что вероятность выполнения равенства (15) для всех пар запросов одинаковая, то общую вероятность получения «1» на выходе алгоритма-различителя можно получить с помощью следующего выражения:

$$P_2(LM) = 1 - (1 - \frac{1}{2^n - 1})^{2^{\frac{k(k-1)}{2}}}, \qquad (21)$$

где k – количество запросов. Вероятность определяется по формуле сложения для совместимых событий, т.к. одновременно несколько пар могут удовлетворять заданному равенству. Общее количество возможных пар составляет $C_k^2 = \frac{k(k-1)}{2}$.

Формула (21) является аппроксимационным значением, имеющим минимальную погрешность при количестве запросов, значительно меньшим мощности множества открытых (шифрованных) текстов.

Точное значение вероятности можно получить с помощью формулы

$$P_3(LM) = 1 - \prod_{i=0}^{k-2} (1 - \frac{1}{2^n - 1 - i})^{2(k-(i+1))}. \qquad (22)$$

Это связано с тем, что пары запросов не являются независимыми и равновероятными. Однако отличия в вероятностях минимальны, поэтому для упрощения целесообразно пользоваться аппроксимированным значением (21). Зависимости между запросами рассматривались в нашей предыдущей работе [2].

Для случайной перестановки вероятность выполнения равенства (15) при $k = 2$ запросах равна

$$P(PRP) = \frac{2^n}{2^{2n} - 1}.$$

Вычитание единицы из знаменателя обусловлено тем, что в перестановке при разных входных значениях не повторяются выходные значения. Соответственно, $W_i \neq W_j$ и невозможено выполнение условия $W_i^L \oplus W_j^L = W_i^R \oplus W_j^R = 0$. Поскольку правая часть выражения (15) никогда не равна нулю (по условию для линейного ортоморфного преообразования), то вероятность увеличивается.

Поскольку правая часть выражения (15) никогда не равна нулю (по условию, для цепи Фейстеля в качестве линейной функции), то невозможен один из неподходящих вариантов, т.е. вероятность увеличивается.

Если принять, что вероятность выполнения равенства (15) для всех пар запросов одинаковая, то общую вероятность получения «1» на выходе алгоритма-различителя для случайной перестановки можно получить с помощью выражения:

$$P_2(LM) = 1 - (1 - \frac{2^n}{2^{2n} - 1})^{\frac{k(k-1)}{2}}.$$

Способ нахождения такой вероятности аналогичен описанному выше. Точное значение вероятности можно получить используя следующее выражение:

$$P_3(LM) = 1 - \prod_{i=0}^{k-2} (1 - \frac{2^n}{2^{2n} - 1 - i \cdot 2^n})^{k-(i+1)}. \qquad (23)$$

Доказательство этой формулы уже приводилось в нашей предыдущей статье [2] и здесь опускается.

Преимущество алгоритма-различителя находится как модуль разности значений (22) и (23):

$$Adv(LM, PRP) = |\prod_{i=0}^{k-2} (1 - \frac{1}{2^n - 1 - i})^{2(k-(i+1))} -$$
$$- \prod_{i=0}^{k-2} (1 - \frac{2^n}{2^{2n} - 1 - i \cdot 2^n})^{k-(i+1)}| \leq$$
$$\leq |(1 - \frac{1}{2^n - 1})^{k(k-1} - (1 - \frac{2^n}{2^{2n} - 1})^{\frac{k(k-1)}{2}}|.$$

Доказательство окончено.

Аналогично предыдущему алгоритму, увеличение количества запросов до определенного порогового значения приводит к возрастанию вероятности различения. Дальнейшие запросы снижают эффективность различения.

На рис. 8 и 9 приведен график зависимости вероятности (ось ординат) различения для $n = 8$ и $n = 16$ от количества входных запросов (ось абсцисс).

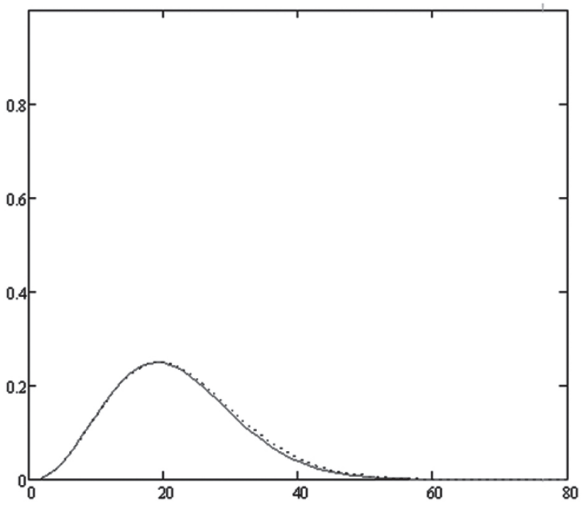


Рис. 8. Зависимость вероятности различения от количества входных запросов для $n = 8$

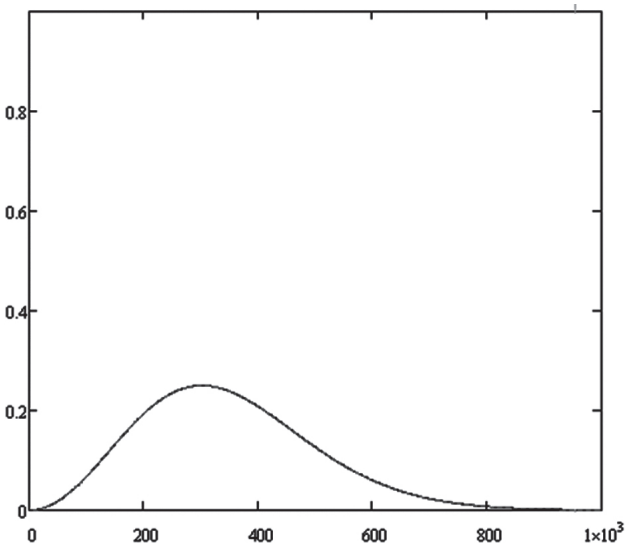


Рис. 9. Зависимость вероятности различения от количества входных запросов для $n = 16$

Как следует из графиков, для выбранных параметров с помощью описанного алгоритма-различителя можно достичь вероятности различения порядка $P_{\max} \approx 0.3$. При этом требуется порядка $\sqrt{2^n}$ выбранных открытых текстов.

Рассматриваемый алгоритм имеет практически такую же эффективность, как и предыдущий для 3-раундовой схемы Лей-Месси со случайными функциями в качестве раундовых преобразований.

ВЫВОДЫ

Использование модели идеального блочного шифра как случайной перестановки позволяет получить численную оценку эффективности высокоуровневой конструкции алгоритма шифрования, в рассмотренном случае – схемы Лей-Месси.

Для исключения влияния свойств конкретной цикловой функции в качестве раундового преобразования целесообразно брать случайную функцию или случайную перестановку. Полученные результаты позволяют точно оценить верхнюю границу эффективности (преимущества) произвольного алгоритма-различителя для 3 раундовой схемы Лей-Месси.

Для рассмотренных конкретных алгоритмов выведены точные значения преимущества, определен метод расчета оптимального количества запросов, при котором преимущество будет максимальным.

Для алгоритмов различения схемы Лей-Месси на основе случайных функций и случайных перестановок существует конкретное значение количества запросов, на котором преимущество будет максимальным.

В то же время, для алгоритма различения цепи Фейстеля на основе случайных перестановок увеличение количества запросов непрерывно ведет к максимизации преимущества.

Дополнительные аппроксимационные соотношения позволяют значительно упростить расчет вероятностей различения и преимущества алгоритмов-различителей с высокой точностью.

Критерий эффективности высокоуровневой конструкции блочного шифра на основе сравнения сложности различения 3-раундового преобразования позволяет сделать вывод о большей эффективности схемы Лей-Месси по сравнению с цепью Фейстеля.

Литература

[1] *Vaudenay S.* On the Lai-Massey Scheme / Technical report LIENS-99-3, Ecole Normale Superieure, 1999.

[2] *Р.В. Олейников, Д.С. Кайдалов.* Уточнение эффективности различения цепи Фейстеля и случайной перестановки. Радиотехника / вып. 167, Харьков: ХНУРЭ, 2011 г., стр. 190-202.

Поступила в редколлегию 11.03.2012

Олейников Роман Васильевич, кандидат технических наук, доцент кафедры БИТ ХНУРЭ. Область научных интересов: анализ и синтез симметричных криптографических преобразований

Кайдалов Дмитрий Сергеевич, аспирант кафедры БИТ ХНУРЭ. Область научных интересов: анализ стойкости блочных симметричных шифров.

УДК 621.391:519.2:519.7

Оцінка складності розрізнення схеми Лей-Мессі та випадкової перестановки / Р.В. Олійников, Д.С. Кайдалов // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 152–159.

Виконаний аналіз ефективності трьохраундової схеми Лей-Мессі – високорівневої конструкції симетричного блокового шифра. Оцінка отримана на основі визначення складності виконання атаки з обраними відкритими текстами, яка спрямована на розрізнення конструкції криптографічного перетворення і моделі ідеального шифра – випадкової перестановки.

Ключевые слова: блочний шифр, схема Лей-Мессі, випадкова перестановка.

Іл. 9. Бібліогр.: 2 назв.

UDC 621.391:519.2:519.7

Evaluation of complexity of distinguishing the Lai-Massey scheme and random permutation / R.V. Oliynykov, D.S. Kaidalov // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 152–159.

An efficiency analysis of the 3-round Lai-Massey scheme as a high level construction of a symmetric block cipher is performed. The evaluation is based on the complexity estimation of a chosen plaintext attack directed at distinguishing the construction of cryptographic transformation and a model of the ideal cipher as a random permutation. The upper bound of advantage for an arbitrary algorithm-distinguisher and precise values of advantage for two methods are grounded.

Keywords: block cipher, Lai-Massey scheme, random permutation.

Fig. 9. Ref.: 2 items.

УДК 621. 391:519.2:519.7

О СТОЙКОСТИ БЛОЧНЫХ ШИФРОВ С RIJNDAEL-ПОДОБНЫМИ ПРЕОБРАЗОВАНИЯМИ К ИНТЕГРАЛЬНЫМ АТАКАМ

В.И. РУЖЕНЦЕВ

Работа посвящена исследованию особенностей организации интегральных атак на различные варианты шифров с rijndael-подобными преобразованиями, выполняется сравнение этих шифров по стойкости к интегральному криптоанализу, уточняются опубликованные ранее результаты относительно стойкости блочных шифров.

Ключевые слова: блочный шифр, интегральный криптоанализ, rijndael-подобные преобразования, *n*-цикловый интеграл.

ВВЕДЕНИЕ

Интегральный криптоанализ является одним из наиболее эффективных видов нападения на самый распространенный в мире шифр rijndael [1] (и его вариант – AES [2]) с уменьшенным количеством циклов. Распространенность данного шифра привела также к частому использованию его элементов в других шифрах. Так, например, все алгоритмы шифрования, которые были представлены в национальном конкурсе блочных симметричных шифров [3], использовали rijndael-подобные преобразования. В наших работах [4-6] проводился анализ возможности организации, в том числе, и интегральной атаки на некоторые из шифров – участников конкурса [3]. Целью настоящей работы является, с одной стороны, изложение особенностей организации интегральных атак на различные варианты шифров с rijndael-подобными преобразованиями, с другой стороны, сравнение этих шифров по стойкости к интегральному криптоанализу.

1. ОБЩАЯ ХАРАКТЕРИСТИКА АТАКИ. ИСПОЛЬЗУЕМЫЕ ОБОЗНАЧЕНИЯ

Интегральная атака на блочные симметричные шифры относится к классу атак на цикловую функцию, и для ее реализации необходимо иметь достаточное множество криптограмм, полученных при зашифровании подобранных открытых текстов на одном и том же секретном ключе.

Интегральной атака названа, потому что в атаке рассматривается прохождение через преобразования шифра суммы состояний. Здесь под различными состояниями понимаются некоторые промежуточные значения блоков преобразуемых данных в процессе их зашифрования. Подобно тому, как в дифференциальном криптоанализе производится “транспортирование” разности через преобразования шифра, в данной атаке через циклы шифра проводится значение суммы состояний из некоторого множества.

Если имеется возможность с высокой вероятностью предсказать значение некоторых битов суммы состояний после r циклов шифрования, то это означает, что может быть организована интегральная атака на (r+1)-цикловый шифр. В ходе атаки перебираются возможные подключи последнего цикла и для каждого варианта производится дешифрование одного цикла для всего множества имеющихся криптограмм. Если в результате суммирования информационных блоков, полученных при одноцикловом дешифровании, на известных позициях будет получено нужное значение, то с высокой вероятностью проверяемая часть подключа последнего цикла является верной. Более подробно особенности организации этого вида атак изложены в [7].

Для дальнейшего описания особенностей организации интегральных атак на различные шифры удобно будет воспользоваться обозначениями, введенными в работе [7]. Интеграл первого порядка обычно состоит из 2^8 состояний. Будем использовать следующие обозначения для отдельных байтов:

A – байт «all» - в каждом из состояний на этой позиции уникальное, отличное от остальных состояний значение.

C – байт «constant» - в каждом из состояний на этой позиции идентичное значение.

S – байт «sum» - сумма по XOR значений на этой позиции для всех состояний равно 0.

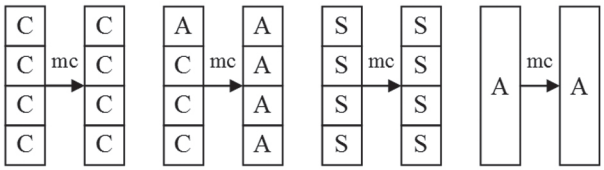
? – про байт ничего не известно.

Интеграл порядка d состоит из 2^{8d} состояний. Байты «C», «S» и «?» имеют прежние значения. Байты «A» имеют верхний и нижний индексы: верхний обозначает порядок интеграла, нижний индекс указывает на то, что конкатенация всех слов (байтов) с одинаковым нижним индексом будет иметь уникальное $8d$-битное значение для каждого состояния. Если все d таких слова (байта) стоят в блоке рядом, то будем обозначать их на рисунках одной общей ячейкой, обозначенной символом «A».

Относительно интегралов разных порядков следует заметить, что при равном числе циклов более эффективен интеграл меньшего порядка, так как его использование в атаке требует меньших вычислительных затрат. Поэтому в дальнейшем для анализируемых шифров будем приводить лишь самые эффективные из найденных интегралов, то есть такие, которые покрывают максимальное количество циклов и обладают минимальным порядком.

2. ИНТЕГРАЛЬНЫЕ СВОЙСТВА RIJNDAEL-ПОДОБНЫХ ПРЕОБРАЗОВАНИЙ

Напомним как «A», «C» и «S» байты проходят через основные rijndael-подобные преобразования. ByteSub или подстановка 8 в 8 битов не меняет байты «A» и «C». Сложение с ключом по XOR не меняет байты «A», «C» и «S». Основные правила прохождения через MixColumn представлены на рис. 1.



а *б* *в* *г*

Рис. 1

При этом следует заметить, что поскольку байты «A» и «C» обладают свойствами байтов «S», то правило на рис. 1, *в* будет применимо, например, и для входной комбинации AACC; результат будет – SSSS. Правило на рис. 1, *г* показывает, что если на вход MixColumn подавать все возможные 32-битные значения, то все возможные значения будут получены и на выходе.

В табл. 1 представлены известные правила изменения байтов при прохождении через операцию XOR, которая часто встречается в схемах Фейстеля и схемах Лея-Месси.

Таблица 1

⊻OP	A	X	Σ	?
A	Σ	A	Σ	?
X	A	X	Σ	?
Σ	Σ	Σ	Σ	?
?	?	?	?	?

В качестве базового циклового преобразования для шифров будем использовать преобразование SL, схема которого представлена на рис. 2 .

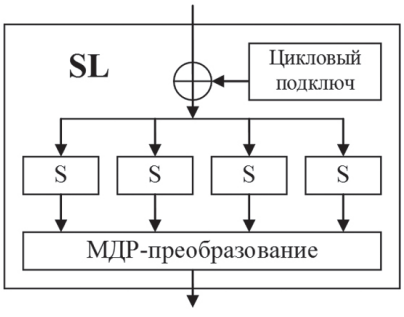


Рис. 2

3. АНАЛИЗ ШИФРОВ, ИСПОЛЬЗУЮЩИХ SL-ПРЕОБРАЗОВАНИЕ

Схема SPN.

Рассмотрим шифр с SL-преобразованием в качестве цикловой функции.

В SPN схеме одно SL-преобразование следует за другим. В такой схеме имеется 2-цикловый интеграл 1-го порядка, на входе которого один активный байт, а на выходе все байты «S» (см. рис. 3).

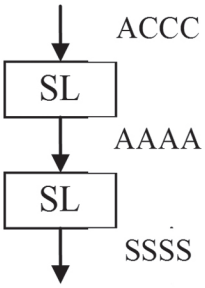


Рис. 3

Такой интеграл позволяет организовать атаку на 3-цикловый шифр.

Интегралы более высоких порядков для данной схемы преобразований не будут более эффективными, так как интеграл 4-го порядка потребует 2^{32} открытых текстов, а это есть полное множество открытых текстов – в этом случае эффективнее словарная атака (построение «словаря» для полного множества входных текстов).

Схема Фейстеля.

Интегральные атаки на фейстель-подобные шифры описаны в работах [4,8].

На рис. 4 представлен 4-цикловый интеграл 1-го порядка, который позволяет организовать атаку на такой шифр с 5 циклами.

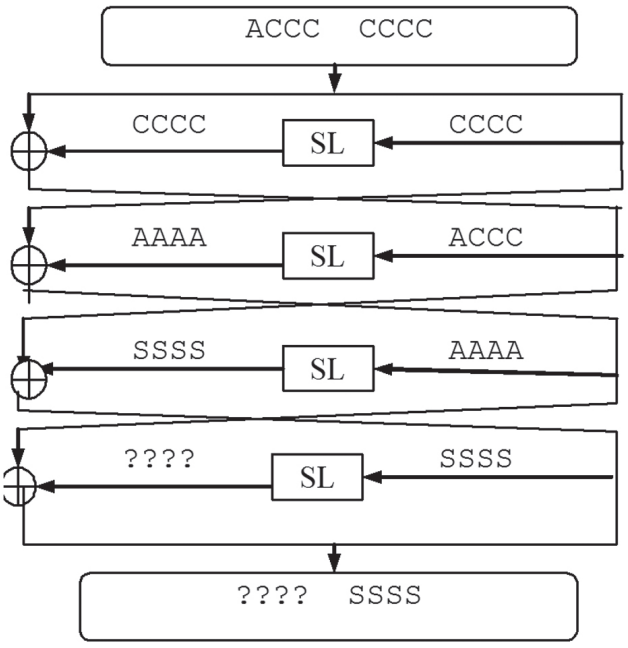


Рис. 4

Интегралов более высоких порядков, покрывающих большее число циклов найдено не было.

Схема Лея-Месси. Особенности организации атак на шифры с использованием схемы Лея-Месси обсуждаются в [5,9].

На рис. 5 представлен 2-цикловый интеграл, который позволяет организовать атаку на такой шифр с 3 циклами.

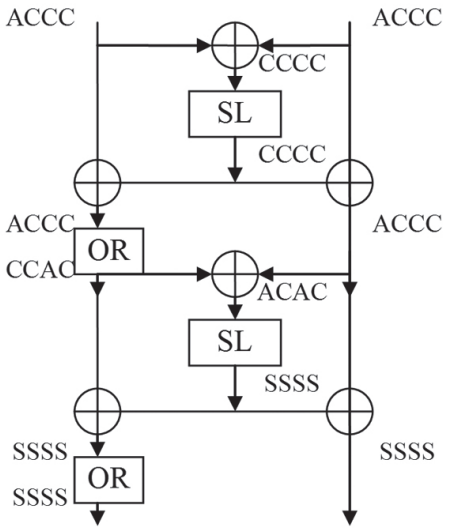


Рис. 5

В рассматриваемом шифре перед первым циклом нет никаких операций, поэтому есть возможность обхода правила сложения по XOR байтов «A» из табл. 1. Подавая на позиции «A» в левом и правом полублоке синхронно одинаковые значения можно инициировать получение при сложении байта «C», что и изображено на рис. 5.

В шифре «Мухомор», который анализировался в [5], присутствует начальное забеливание, что делает описанный интеграл неприменимым. Наиболее эффективными для данного шифра остаются интегралы, представленные в [5].

Интегралов более высоких порядков, покрывающих большее число циклов для рассматриваемого варианта шифра, найдено не было.

Учитывая то, что по сравнению со схемой SPN схемы Фейстеля и Лея-Месси имеют в 2 раза больший размер блока, среди рассмотренных вариантов использования SL-преобразования предпочтительнее остальных выглядит схема Лея-Месси. Главная причина этого, на наш взгляд, заключается в достаточно большом количестве операций XOR между подблоками (3 в одном цикле плюс 1 в операции ORT), что затрудняет прохождение через преобразования байтов «A».

4. ВАРИАНТЫ СХЕМЫ SPN

В этом подразделе уточняются представленные в работе [6] данные о стойкости к интегральной атаке rijndael-подобных шифров, построенных с использование предложенного в [10] подхода. К таким шифрам, в частности, относится алгоритм «Калина» [11].

Кратко изложим суть предложенного в [10] подхода.

Анализ известных БСШ показал, что существуют два способа использования МДР-преобразований при построении операций рассеивания SPN-шифров. При первом способе используется МДР-преобразование, которое покрывает весь блок. В этом случае блок представляется в виде вектора, а преобразование заключается в умножении этого вектора на квадратную МДР-матрицу соответствующего размера. Схема такого преобразования для размера блока m байтов представлена на рис. 6, *а*. Таким строением линейных преобразований обладают шифры Shark, Khazad и рассмотренные в разделе 3 варианты SPN шифров. В соответствии с основным свойством МДР-преобразований, каждая 2-цикловая дифференциальная или линейная характеристика для этих шифров будет содержать не менее m+1 активных S-блоков. Основным недостатком данного способа построения линейных преобразований является большой размер используемой МДР-матрицы, что обычно приводит к повышенной сложности вычислений.

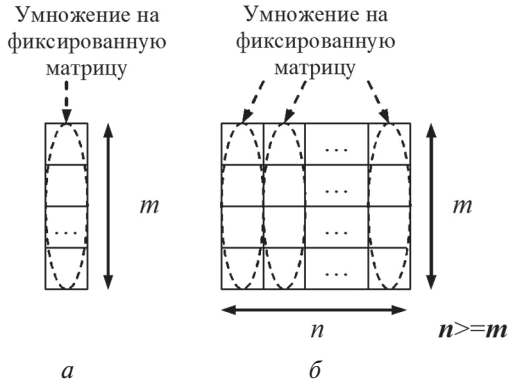


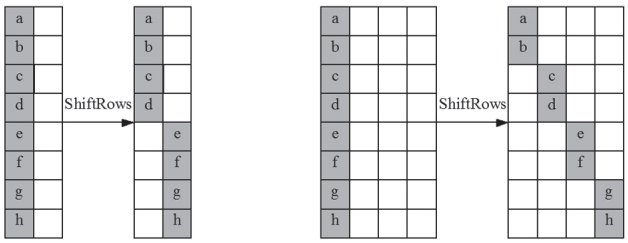
Рис. 6. Известные схемы построения линейных операций с использованием МДР-преобразований

Во втором случае используется “rijndael-подобная” структура линейных преобразований, то есть, блок разбивается на n векторов по m байтов каждый (общий размер блока $n \times m$ байтов), при чем, n больше или равен m (см. рис. 6, *б*).

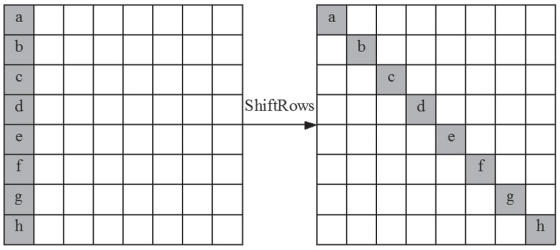
Линейное преобразование заключается в том, что каждый m-байтный вектор, называемый колонкой, умножается на МДР-матрицу размером $m \times m$ байтов, после чего выполняется байтовая перестановка, в ходе которой байты каждой колонки распределяются по одному байту в каждую колонку. Аналог первой части операции рассеивания – процедура MixColumns в Rijndael, аналог байтовой перестановки – процедура ShiftRows. Подобное строение линейных преобразований используется также в шифрах Square, Crypton, Anubis.

В [10] обсуждался вариант линейных преобразований, когда $m \geq n$. По такой схеме построены линейные преобразования шифра «Калина» [11]. В этом шифре $m = 8$, n меняется от 2 до 8 в зависимости от размера блока. В ходе операции MixColumns выполняется умножение на фиксированную матрицу размером 8×8 байтов, а порядок перестановки байтов в ходе операции ShiftRows представлен на рис. 7.

Уточнения касаются возможности организации интегральной атаки на варианты шифра с одним дополнительным циклом по сравнению с представленными в [6] результатами.



а – 128-битный блок *б* – 256-битный блок



в – 512-битный блок

Рис. 7

По аналогии с 4-цикловым интегралом 4-го порядка для шифра Rijndael, предложенного в [12], для рассматриваемого шифра также существует 4-цикловый интеграл порядка равного количеству байтов в колонке. На вход подаются 2^{64} состояний, которые имеют уникальное 64-битное значение в 8 байтах, отмеченных серым цветом в начальной позиции на рис. 8.

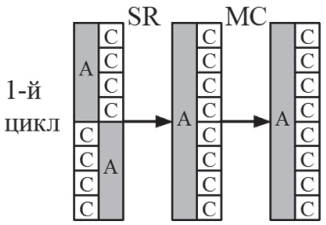


Рис. 8

Как видно из рис. 8, после операции ShiftRows эти 8 байтов оказываются в одной колонке. А в соответствии с правилом, представленным на рис. 1,г, после операции MixColumn будет получен такой же набор состояний. Этот набор состояний эквивалентен 2^{32} наборам состояний, соответствующим входной позиции интеграла, который был предложен в [6] и представлен на рис. 9. (Каждый из 2^{32} наборов состояний отличается от остальных значениями в четырех байтах «С» первой колонки.)

В итоге, мы получаем 4-цикловый интеграл 8-го порядка, который позволяет организовать атаку на 5-цикловый шифр с подобной структурой преобразований со сложностью около 2^{65} операций шифрования.

Следует заметить, что подобный 4-цикловый интеграл 8-го порядка (порядок равен количеству байтов в колонке) существует и для других вариантов SPN шифров, использующих как схему линейных преобразований, представленную на рис. 6,б, так и схему, предложенную в работе [10].

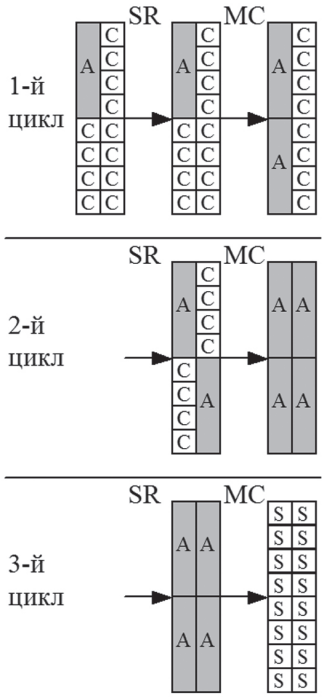


Рис. 9

ВЫВОДЫ

В работе выполнено исследование стойкости к интегральному криптоанализу шифров с rijndael-подобными цикловыми преобразованиями построенных с использованием схем SPN, Фейстеля и Лея-Месси. Сделан вывод о том, что более высокий уровень безопасности обеспечивает схема Лея-Месси за счет большего, по сравнению с другими схемами, количества дополнительных операций XOR.

С учетом материалов работы [12], уточнены опубликованные ранее результаты о стойкости шифра «Калина» к интегральной атаке из [6]. Описано как может быть организована атака на варианты шифра с 5 циклами, что на 1 цикл больше, чем описано в [6]. Алгоритм с полным набором циклов обеспечивает защищенность от интегральной атаки.

Литература

[1] *J. Daemen, V. Rijmen.* AES Proposal Rijndael, AES Round 1 Technical Evaluation CD1: Documentation, National Institute of Standards and Technology, Aug 1998. See http://www.nist.gov/aes.

[2] National Institute of Standards and Technology “Advanced encryption algorithm (AES) development.” // FIPS 197, U.S. Department of Commerce, Nov. 2001.

[3] Офіційний ресурс департамента спеціальних телекомунікаційних систем та захисту інформації: «Положення про проведення відкритого конкурсу криптографічних алгоритмів», 2006. Доступно по адресу http://www.dstszi.gov.ua/dstszi/control/ru/publish/article/.

[4] *Долгов В.И., Головашич С.А., Руженцев В.И.* Криптостойкость шифра “Торнадо” // Радиотехника. 2003. № 134. С. 81-88.

[5] *Горбенко І.Д., Долгов В.І., Руженцев В.І. Олійников Р.В., Михайленко М.С.* Криптостійкість шифру

"Мухомор" // Прикладная радиоэлектроника. Тематический выпуск, посвященный проблемам обеспечения безопасности информации. Харьков. Том 6, №2, 2007 г. С. 186-194.

[6] *Горбенко І.Д., Долгов В.І., Руженцев В.І. Олійников Р.В., Михайленко М.С.* Криптостійкість шифру "Калина" // Прикладная радиоэлектроника. Тематический выпуск, посвященный проблемам обеспечения безопасности информации. Харьков. Том 6, №2, 2007 г. С. 217-229.

[7] *L. R. Knudsen*. Integral Cryptanalysis, NESSIE internal report NES/DOC/UIB/WP5/015/1, 2001.

[8] *Y.-J. Li, W.-L. Wu, L.-T. Zhang, L. Zhang,* Improved Integral Attacks on Reduced Round Camellia, IACR Eprint archive, 2011. available from http://eprint.iacr.org/2011/163.

[9] *W. Wenling, Z. Wentao, F. Dengguo,* Improved Integral Cryptanalysis of FOX Block Cipher, IACR Eprint archive, 2005. available from http://eprint.iacr.org/2005/292.

[10] *V. Ruzhentsev, R. Oliynykov,* Properties of Linear Transformations for Symmetric Block Ciphers on the basis of MDS-codes // Proceedings of the 6th International Conference on Network Architecture and Information System Security SAR-SSI 2011, pp. 193-196. La Rochelle, France.

[11] *Горбенко І.Д., Долгов В.І., Олійников Р.В., Руженцев В.І. та ін.* Перспективний блоковий симетричний шифр "Калина" – основні положення та специфікація // Прикладная радиоэлектроника. Тематический выпуск, посвященный проблемам обеспечения безопасности информации. Харьков. Том 6, №2, 2007 г. С. 195-208.

[12] *N. Ferguson, J. Kelsey, S. Lucks, B. Schneier, M. Stay, D. Wagner, and D. Whiting.* Improved Cryptanalysis of Rijndael. FSE 2000, LNCS 1978, pp. 213-230, Springer-Verlag, 2001.

Поступила в редколлегию 26.03.2012

Руженцев Виктор Игоревич, кандидат технических наук, доцент кафедры БИТ ХНУРЭ. Область научных интересов: симметричные криптоалгоритмы, криптоанализ.

УДК 621. 391:519.2:519.7

Про стійкість блокових шифрів з rijndael-подібними перетвореннями до інтегральних атак / В.І. Руженцев // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 160–164.

Робота присвячена дослідженню особливостей організації інтегральних атак на різні варіанті шифрів з rijndael-подібними перетвореннями, виконується порівняння цих шифрів за рівнем стійкості до інтегрального криптоаналізу, уточнюються деякі з раніше опублікованих результатів стосовно стійкості блокових шифрів.

Ключові слова: блоковий шифр, інтегральний криптоаналіз, rijndael-подібні перетворення, *n*-цикловий інтеграл.

Табл. 1. Іл. 9. Бібліогр.: 12 назв.

UDC 621. 391:519.2:519.7

On the resistance of block ciphers with Rijndael-like transformations to integral attacks / V.I. Ruzhentsev // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 160–164.

The paper is dedicated to studying perculiarities of organizing integral attacks on different variants of ciphers with rijndael-like transformations. Comparison of these ciphers resistance to integral cryptanalysis is performed. Some of the published earlier results about the ciphers resistance are refined.

Key words: block cipher, integral cryptanalysis, rijndael-like transformations, *n*-round integral.

Tab. 1. Fig. 9. Ref.: 12 items.

УДК 681.3.07

МЕТОД УНІВЕРСАЛЬНОГО ГЕШУВАННЯ ПО РАЦІОНАЛЬНИМ ФУНКЦІЯМ АЛГЕБРАЇЧНИХ КРИВИХ НАД КІЛЬЦЯМИ

А.О. БОЙКО, Г.З. ХАЛІМОВ

Запропоновано метод універсального гешування по раціональним функціям алгебраїчних кривих над кільцями векторів, що на відміну від існуючих методів універсального гешування в полях забезпечує вищу швидкодію і не вразливий до атак спостереження за часом виконання.

Ключові слова: універсальна функція гешування, перетворення над кільцями.

ОГЛЯД СТАНУ

Універсальні геш-функції — це родина геш-функцій, що виконують наступне відображення: $KxM \to H$, де K – множина ключів, M – множина вхідних повідомлень, H – множина геш-значень, для якої виконується для будь-яких $x, y \in M$ наступне .

$$Pr_{h\in H}\{h(x)=h(y)\}=\frac{1}{|H|}. \qquad (1)$$

Алгоритм UMAC [1] є алгоритмом універсального гешування, що виконує гешування по проективним прямим.

UMAC в якості основного кроку використовує функцію поліноміального гешування PolyCW, яка обчислюється за формулою

$$h_x(m)=\sum_{i=0}^{k} m_i x^i \bmod p, \qquad (2)$$

де p – просте число і k – ціле число, $k>0$.

Поліноміальна родина геш-функций PolyCW є універсальною, і імовірність колізії [1]

$$Pr_{h\in H}\{h_x(a)=h_x(b)\bmod p\}=k/p. \qquad (3)$$

Це визначається основною теоремою алгебри, що стверджує, що у полінома степені k може бути не більше k коренів.

У роботах [2-4] було запропоновано методи універсального гешування по проективним кривим, визначеним над полями Галуа.

Недоліком методів гешування, що використовують обчислення над простим полем, є вразливість до атак спостереження за часом виконання. Зазвичай, вхідні блоки повідомлення представляються числами $0 \le m_i \le 2^n - 1$. Однак модуль перетворень P має вигляд $2^n - k$. При обчисленні геш-значення блок $m_i < k$ можна замінити на $m_i + P$ так, що геш-значення не зміниться. Для уникнення такої можливості блок повідомлення, такий що $m_i \ge P - 2$, заміняється парою $\{P-2, P-m_i\}$. Однак для обробки такої пари необхідно витратити в 2 рази більше часу, ніж для обробки "дозволеного" блоку, що і дає інформацію зловмиснику щодо вмісту блоків, які обробляються. Крім того, кожне таке розширення повідомлення приводить до зростання імовірності колізії у відповідності до формули (3).

Можливі шляхи вирішення проблеми:

1) використання проективних кривих, визначених над розширеним двійковим полем;

2) використання проективних кривих, визначених над кільцем, що містить 2^n елементів.

ВИКОРИСТАННЯ ПРОЕКТИВНИХ КРИВИХ, ВИЗНАЧЕНИХ НАД РОЗШИРЕНИМ ДВІЙКОВИМ ПОЛЕМ

Сучасні процесори загального призначення не мають інструкцій, орієнтованих на реалізацію операцій у $GF(2^n)$. Тому при реалізації операцій було використано методи оптимізації, запропоновані у роботі [5]. Однак ці методи вимагають чисельних вибірок з пам'яті по невирівненим адресам, що суттєво знижує швидкодію.

Внаслідок того, що набори інструкцій сучасних процесорів загального призначення не підтримують обчислення у розширеному двійковому полі, то швидкодія методів гешування, що використовують проективні криві, визначені над розширеним двійковим полем, значно гірша за методи, що використовують обчислення у простому полі або в кільці, незважаючи на оптимізації.

Використання проективних кривих, визначених над кільцем з 2^n елементів

Переваги використання кривих над кільцем для гешування:

– висока швидкодія, оскільки всі обчислення виконуються по модулю 2^n, тобто окремої операції приведення по модулю не потрібно, лише складання і множення, що потребує мінімум інструкцій процесора;

– невразливість до атак спостереження за часом виконання.

МЕТОД ПОБУДУВАННЯ УНІВЕРСАЛЬНИХ ГЕШ-ФУНКЦІЙ, ЩО ВИКОРИСТОВУЮТЬ ОБЧИСЛЕННЯ ПО ПРОЕКТИВНИХ КРИВИМ НАД КІЛЬЦЕМ ВЕКТОРІВ

Побудування і властивості кільця векторів

В ході досліджень проективних кривих над кільцем цілих чисел по модулю 2^n виявлено, що такі криві мають дуже мало точок і є ізоморфними проективним прямим, визначеним над тим же кільцем. Тобто, геш-функції, побудовані на основі таких кривих не мають переваг у імовірності колізії над геш-функціями, побудованими на основі обчислень над проективними прямими.

Вперше використання обчислень векторів для побудування груп для криптографічних застосувань (електронного цифрового підпису) було запропоновано у роботі [6]. У роботі [6] запропоновані наступні положення:

1) вектори виду $ae+bi+...+cj$, де $e,i,...j$ – базисні вектори, які також можуть бути представлені у вигляді набору координат $(a,b,...c)$, які є елементами скінченого поля $GF(p)$;

2) операція складання векторів визначається як складання одноіменних координат;

3) операція множення векторів визначається по правилу множення поліномів із урахуванням того, що множення базисних векторів виконується за правилами, заданими таблично;

4) результатом множення базисних векторів є базисний вектор або базисний вектор, помножений на коефіцієнт розтягнення, що обраний спеціальним чином з числа елементів поля $GF(p)$;

5) множина векторів виду $ae+bi+...+cj$ за умови того, що таблиця множення базових векторів має спеціальний вигляд, утворює поле.

Однак обчислення у полі векторів, визначених над простим полем, мають той же недолік, що і самі обчислення у простому полі — можливість атак спостереження за часом виконання. Тому запропоновано для побудування геш-функцій використовувати обчислення у кільці векторів, визначених над кільцем цілих чисел по модулю 2^n (далі позначається як $Z/Z2^n$).

У якості вектора приймається кортеж $\{a,b\}$ з двох елементів $a,b\in Z/Z2^n$. Також цей вектор може бути представлений у вигляді полінома $a+bi$, де $a,b\in Z/Z2^n$, а i — базисний вектор, для якого вірно $i^2=-1 \bmod 2^n$.

Додавання векторів визначено у відповідності до [6] як

$$\begin{gathered}(a+bi)+(c+di)= \\ \left((a+c)\bmod 2^n+\left((b+d)\bmod 2^n\right)i\right).\end{gathered} \quad (4)$$

Множення векторів визначено як

$$\begin{gathered}(a+bi)(c+di)= \\ \left((ac-bd)\bmod 2^n\right)+\left((ad+bc)\bmod 2^n\right)i.\end{gathered} \quad (5)$$

Операції додавання і множення векторів формально співпадають з відповідними операціями над комплексними числами з тією лише різницею, що у випадку векторів всі обчислення здійснюються по модулю 2^n (координати вектора належать $Z/Z2^n$).

Твердження. Множина векторів, що мають 2 координати з $Z/Z2^n$ і $i^2=-1\bmod 2^n$, над якою визначені операції додавання і множення векторів, є комутативним кільцем.

Доведення.

Комутативним кільцем називається множина R, над якою задані бінарні операції додавання "+" і множення "*" такі, що:

$r_1+(r_2+r_3)=(r_1+r_2)+r_3$ для всіх $r_1,r_2,r_3\in R$;

$r_1*(r_2*r_3)=(r_1*r_2)*r_3$ для всіх $r_1,r_2,r_3\in R$;

$r_1*r_2=r_2*r_1$ і $r_1+r_2=r_2+r_1$ для всіх; $r_1,r_2\in R$

існує такий елемент $O\in R$, що для усіх $r\in R$ виконується $r+O=O+r=r$;

існує такий елемент $e\in R$, що для усіх $r\in R$ виконується $r*e=e*r=r$;

для усіх $r\in R$ існує $r'\in R$ таке, що $r+r'=O$.

Асоціативність операції додавання доводиться як

$$\begin{gathered}\big((a+bi)+(c+di)\big)+(e+fi)= \\ \big(((a+c)+e)+((b+d)+f)i\big)= \\ \big((a+(c+e))+(b+(d+f))i\big)= \\ (a+bi)+\big((c+di)+(e+fi)\big)\end{gathered}.$$

Асоціативність операції множення доводиться як

$$\begin{gathered}\big((a+bi)*(c+di)\big)*(e+fi)= \\ \big((ac-bd)+(bc+ad)i\big)(e+fi)= \\ \big((ac-bd)e-(bc+ad)f\big)+ \\ +\big((ac-bd)f+(bc+ad)e\big)i= \\ (ace-bde-bcf-adf)+ \\ +(acf-bdf+bce+ade)i \\ (a+bi)*\big((c+di)*(e+fi)\big)= \\ (a+bi)*\big((ce-df)+(cf+de)i\big)= \\ \big(a(ce-df)-b(cf+de)\big)+ \\ +\big(b(ce-df)+a(cf+de)\big)i= \\ (ace-adf-bcf-bde)+ \\ +(bce-bdf+acf+ade)i\end{gathered}$$

Результат однаковий з точністю до порядка членів. Оскільки у $Z/Z2^n$ додавання комутативне, то можна прийняти, що результати однакові.

Комутативність додавання доводиться як

$$(a+bi)+(c+di)=(a+c)+(b+d)i.$$

Оскільки у $Z/Z2^n$ додавання комутативне, то

$$\begin{gathered}(a+c)+(b+d)i=(c+a)+(d+b)i= \\ (c+di)+(a+bi)\end{gathered}.$$

Комутативність множення доводиться як

$$(a+bi)(c+di)=(ac-bd)+(bc+ad)i,$$

$$(c+di)(a+bi)=(ca-db)+(cb+da)i.$$

Оскільки у $Z/Z2^n$ додавання комутативне, то $(ca-db)+(cb+da)i=(ac-bd)+(bc+ad)i$, звідки $(a+bi)(c+di)=(c+di)(a+bi)$.

Нульовим елементом у кільці векторів є {0, 0} (або $0+0i$ у поліноміальному представленні):

$$(a+bi)+(0+0i)=(0+0i)+(a+bi)=$$
$$(a+0)+(b+0)i=a+bi.$$

Одиничним елементом у кільці векторів є {1, 0} (або $1+0i$ у поліноміальному представленні):

$$(a+bi)(1+0i)=(1a-b0)+(1b+a0)i=$$
$$(a-0)+(b+0)i=a+bi.$$

Для вектора $a+bi$ аддитивно оберненим буде вектор $(2^n-a)+(2^n-b)i$, оскільки

$$(a+bi)+((2^n-a)+(2^n-b)i)=$$
$$=(a+2^n-a)+(b+2^n-b)i=2^n+2^n i=0+0i.$$

Твердження доведено.

Тут і далі, коли згадується кільце векторів, то мається на увазі саме кільце векторів, визначених над кільцем цілих чисел по модулю 2^n.

Твердження. Вектори, у яких одна з координат парна, а інша — непарна, утворюють циклічну мультиплікативну групу, у якій відсутні дільники нуля (оскільки сам нульовий вектор {0, 0} не належить до цієї групи).

Доведення. Як було раніше вказано, операції над векторами з 2-ма координатами з $Z/Z2^n$ і $i^2=-1 \bmod 2^n$ відповідають операціям над комплексними числами. Розглянемо матричне представлення комплексного числа. Комплексне число $a+bi$ може бути представлене у вигляді $\begin{pmatrix} a & b \\ -b & a \end{pmatrix}$. Детермінант такої матриці визначається як $\det\begin{pmatrix} a & b \\ -b & a \end{pmatrix}=a^2+b^2$. Отже, якщо координата a або b непарна, а інша число координата парна, то детермінант такої матриці зажди непарний:

$$(2k_1+1)^2+(2k_2)^2=4k_1^2+4k_1+1+4k_2^2=$$
$$2(2k_1^2+2k_1+2k_2^2)+1.$$

Непарні числа по модулю 2^n утворюють циклічну мультиплікативну групу. Наприклад:

$$3^0 \bmod 16=1$$
$$3^1 \bmod 16=3$$
$$3^2 \bmod 16=9$$
$$3^3 \bmod 16=11$$
$$3^4 \bmod 16=1$$

Оскільки

$$\det\left(\begin{pmatrix} a & b \\ -b & a \end{pmatrix}\begin{pmatrix} c & d \\ -d & c \end{pmatrix}\right)=\det\begin{pmatrix} a & b \\ -b & a \end{pmatrix}*\det\begin{pmatrix} c & d \\ -d & c \end{pmatrix},$$

то обчислення детермінанту можна розглядати як гомоморфізм, що відображає мультиплікативну групу матриць виду $\begin{pmatrix} a & b \\ -b & a \end{pmatrix}$ у мультиплікативну групу непарних чисел по модулю 2^n. З цього, а також з того, що мультиплікативна група непарних чисел по модулю 2^n є циклічною, слідує, що мультиплікативна група матриць виду $\begin{pmatrix} a & b \\ -b & a \end{pmatrix}$ також є циклічною. Отже і вектори, у яких одна координата є парною, а інша — непарною, утворюють циклічну мультиплікативну групу. Твердження доведено.

Використання мультиплікативної групи, що має дільники нуля, може погіршити властивості геш-функцій, тому запропоновано використовувати в якості ключів саме вектори, у яких одна з координат парна, а інша — непарна.

Однак сумма двох векторів, що належать циклічній мультиплікативній групі, не належить цій групі, оскільки обидві її координати або парні, або непарні.

Однак можливо при перетворенні блоку повідомлення у вектор можливо робити це таким чином, щоб обидві координати вектора були або парними, або непарними. Тоді усі обчислення повертатимуть результат, що належатиме циклічній мультиплікативній групі.

Однак вектори, що належать групі, не можуть утворити точку на кривій Ферма виду $X^m+Y^m+1=0$, тому що сумма X^m+Y^m+1 належатиме циклічній мультиплікативній групі, тоді як $0+0i$ до неї не належатиме.

Тому запропоновано використовувати криву виду $X^m+Y^m+2=0$, або в узагальненому вигляді $X^m+Y^m+\{2s,2t\}=0$.

ВЛАСТИВОСТІ ГЕШ-ФУНКЦІЙ, ЩО ВИКОРИСТОВУЮТЬ ОБЧИСЛЕННЯ ПО ПРОЕКТИВНИМ КРИВИМ НАД КІЛЬЦЕМ ВЕКТОРІВ

У зв'язку з тим, що теорія проективних кривих над кільцями ще не розроблена, усі дослідження проводилися шляхом виконання обчислювальних експериментів з кривими над невеликими кільцями.

Розглядалась можливість використання кривих Ферма виду $X^m+Y^m+const=0$ над кільцем векторів виду $x+yi$, де $0\le x,y<2^n$, а $i^2=-1 \bmod 2^n$.

Під час досліджень актуальними питаннями були:

1) яку кількість точок має крива Ферма над кільцем векторів і як ця кількість змінюється в залежності від розміру кільця;

2) які колізійні властивості мають геш-функції, побудовані на кривих Ферма над кільцем векторів.

При дослідженні кривих Ферма над кільцем векторів з обмеженням, заданим у твердженні (1), найкращі результати показали криві виду $X^m+Y^m+\{2^{n-1},2^{n-1}\}=0$, де $m=4k+2$ з цілим $k>0$, 2^n — модуль перетворень у кільці, якому належать координати вектора. У всіх проведених

експериментах такі криві завжди мали 2^{n+2} точок. Дослідження виконувалося шляхом побудування всіх можливих точок на кривій перебором.

Дослідження колізійних властивостей геш-функції проводилось наступним чином:

1) було побудовано матрицю значень усіх раціональних функцій степені не вище k від координат точок кривої x і y, розмір матриці $k(k+1)/2-1\,x\,N$, де k — максимальна степінь раціональної функції, а N — кількість точок на кривій;

2) випадковим чином було згенеровано $k(k+1)/2-1$ блоків повідомлення від a_1 до $a_{k(k+1)/2-1}$;

3) за формулою (6) було обчислено N значень від s_1 до s_N, що за фізичним змістом представлять собою множину усіх значень геш-функції при фіксованому повідомленні $\{a_1 \| ... \| a_{k(k+1)/2-1}\}$ і всіх можливих значеннях ключів від (x_1, y_1) до (x_N, y_N);

$$\begin{pmatrix} x_1 & y_1 & x_1^2 & x_1y_1 & y_1^2 & \dots & x_1^k y_1^k \\ x_2 & y_2 & x_2^2 & x_2y_2 & y_2^2 & \dots & x_2^k y_2^k \\ x_3 & y_3 & x_3^2 & x_3y_3 & y_3^2 & \dots & x_3^k y_3^k \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ x_N & y_N & x_N^2 & x_Ny_N & y_N^2 & \dots & x_N^k y_N^k \end{pmatrix} * \begin{pmatrix} a_1 \\ a_2 \\ \dots \\ a_{k(k+1)/2-1} \end{pmatrix} = \begin{pmatrix} s_1 \\ s_2 \\ \dots \\ s_N \end{pmatrix}; \quad (6)$$

4) у множині значень від s_1 до s_N було знайдено значення, що зустрічається найчастіше і підраховано кількість появ цього значення;

5) отримане на попередньому кроці число було занесене до відповідного списку;

6) кроки 2-5 було повторено 10^6 разів;

7) значення k змінювалося від 2 до 6, N приймало значення $2^8, 2^{10}, 2^{12}, 2^{14}, 2^{16}$.

Отримані результати внесені у табл. 1.

Таблиця 1

n	Макс степінь раціональної функції, k	Модуль 2^n	Розмір кільця векторів	$N_т$ кількість точок на кривій 2^{2n+2}	$N_{макс}$ максимальна кількість однакових значень	Імовірність коллізії $N_{макс}/N_т$	Примітка
3	2	8	32	$256=2^8$	64	0,25	1/4
3	3	8	32	$256=2^8$	64	0,25	1/4
3	4	8	32	$256=2^8$	64	0,25	1/4
3	5	8	32	$256=2^8$	32	0,125	1/8
3	6	8	32	$256=2^8$	192	0,75	3/4
4	2	16	128	$1024=2^{10}$	96	0,09375	3/32
4	3	16	128	$1024=2^{10}$	96	0,09375	3/32
4	4	16	128	$1024=2^{10}$	96	0,09375	3/32
4	5	16	128	$1024=2^{10}$	64	0,0625	1/16
4	6	16	128	$1024=2^{10}$	768	0,75	3/4
5	2	32	512	$4096=2^{12}$	160	0,03906	
5	3	32	512	$4096=2^{12}$	160	0,03906	
5	4	32	512	$4096=2^{12}$	160	0,03906	
5	5	32	512	$4096=2^{12}$	128	0,03125	1/32
5	6	32	512	$4096=2^{12}$	1568	0,38281	
6	2	64	2048	$16384=2^{14}$	288	0,01758	
6	3	64	2048	$16384=2^{14}$	288	0,01758	
6	4	64	2048	$16384=2^{14}$	288	0,01758	
6	5	64	2048	$16384=2^{14}$	256	0,01562	1/64
6	6	64	2048	$16384=2^{14}$	4224	0,25781	
7	2	128	8192	$65536=2^{16}$	544	0,00830	
7	3	128	8192	$65536=2^{16}$	544	0,00830	
7	4	128	8192	$65536=2^{16}$	544	0,00830	
7	5	128	8192	$65536=2^{16}$	512	0,00781	1/128
7	6	128	8192	$65536=2^{16}$	8352	0,12744	

З табл. 1 видно, що найкращі результати з імовірності колізії досягаються, коли максимальна степінь раціональних функцій дорівнює 5 (повідомлення з 14 блоків) і досягає $\frac{1}{2^n}$, де 2^n – модуль перетворень (розмір кільця, над яким визначено кільце векторів), і одразу різко погіршуються, коли максимальна степінь раціональних функцій досягає 6. Таким чином на повідомленні з 14 блоків геш-функція веде себе як універсальна геш-функція. Із збільшенням розміру кільця векторів колізійна стійкість геш-функцій зростає.

Практичні результати вимірювання швидкодії різних методів універсального гешування наведені у табл. 2. Висока швидкодія методу гешування по проективним кривим над кільцями пояснюється тим, що алгоритми складання і множення векторів є простими і повністю використовують можливості по одночасному виконанню кількох команд у сучасних суперскалярних процесорах.

З таблиці видно, що програш у швидкодії при використанні операцій у $GF\left(2^n\right)$ складає 5-10 раз, що у більшості застосувань є неприпустимим.

В той же час швидкодія алгоритму гешування над кільцями в майже в 4 рази більша ніж гешування по кривим Ферма над квадратичним полем.

Нерозв'язані питання для наступних досліджень:

1) побудова теорії проективних кривих над кільцями, що дозволила б обчислювати число точок на кривій і шукати криві з більшим числом точок, а також уникнути погіршення колізійних властивостей при збільшенні довжини повідомлення;

2) побудова методу оцінки колізійної стійкості геш-функцій, що використовують криві Ферма над кільцями;

3) розробка алгоритму генерації ключів для таких геш-функцій.

ВИСНОВКИ

1. Основним недоліком алгоритмів гешування, що використовують перетворення у простому полі по модулю $2^n - k$ є вразливість до атаки спостереження за часом виконання.

2. Алгоритми гешування, що використовують перетворення у розширеному двійковому полі, не вразливі до атаки спостереження за часом виконання, однак мають значно нижчу швидкодію.

3. Для того, щоб уникнути вразливості до атаки спостереження за часом виконання і зберегти високу швидкодію вперше запропоновано метод універсального гешування, по раціональним функціями проективних кривих над кільцями векторів, які в свою чергу також визначені над кільцями цілих чисел по модулю 2^n.

4. Для практичної перевірки було розглянуто метод гешування по кривим Ферма над кільцями векторів. Найкращі результати з імовірності колізії досягаються, коли максимальна степінь раціональних функцій дорівнює 5 (повідомлення з $5*(5+1)/2-1=14$, блоків), таким чином на повідомленні з 14 блоків геш-функція веде себе як універсальна геш-функція.

Таблиця 2

Метод гешування	Швидкодія (тактів на байт)	Розмір поля (кільця)	Довжина ключа (біт)	Довжина геш-значення	Імовірність колізії
Гешування по проективній прямій $GF(q)$ $q=2^{64}-59$	8,7	$2^{64}-59$	64	64	$\approx 2^{64}$
Гешування по проективній прямій $GF(q)$ $q=2^{32}-5$ $f(t)=t^2-2$	4,8	$(2^{32}-5)^2$	64	64	$\approx 2^{64}$
Гешування по кривій Ферма $GF(q)$ $q=2^{32}-5$	3,7	$2^{32}-5$	64	32	$\approx 2^{64}$
Гешування по проективній прямій $GF\left(\left(\left(2^8\right)^4\right)^2\right)$	41	2^{64}	64	64	$\approx 2^{64}$
Гешування по кривій Ерміта $GF\left(\left(2^8\right)^4\right)$	18	2^{32}	64	32	$\approx 2^{64}$
Гешування по кривим Ферма над кільцем векторів	1,0	2^{128}	256	128	$\approx 2^{64}$

5. Завдяки простим алгоритмам додавання і множення векторів і можливості використання паралельних обчислень всередині цих алгоритмів геш-функції по проективним кривим над кільцями векторів мають дуже високу швидкодію.

6. Нерозв'язаними питаннями залишаються побудова теорії проективних кривих над кільцями і побудова алгоритму генерування ключів для такої геш-функції.

Література

[1] *J. Black, S. Halevi, H. Krawczyk, T. Krovetz, P. Rogaway.* UMAC: Fast and Secure Message Authentication [Електронний ресурс] http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=B1DBCEDF9D4955AF4E565C921F9B38C8?doi=10.1.1.114.7878&rep=rep1&type=pdf.

2] *Халимов Г.З., Котух Е.В.* Универсальное хеширование по кривой Сузуки. Прикладная радиоэлектроника, Том 10, 2011, №2, с.164 – 170.

3] *Халимов Г.З.* Каскадное универсальное хеширование по рациональным функциям алгебраических кривых. Радиотехника, 2011, вып. 166 с. 26-31.

4] *Халимов Г.З.* Универсальное хеширование по максимальным кривым 2-го рода. Тезисы докладов международной конференции "Современные проблемы математики и ее приложения в естественных науках и информационных технологиях", Харьков 2011, с. 191-193.

5] *K Greenan, E. Miller, T. Schwarz.* Optimizing Galois Field Arithmetic for Diverse Processor Architectures and Applications [Електронний ресурс] http://dicc.ucu.edu.uy/publicaciones/mascots08GF.pdf.

6] *Молдовян Н.А.* Группы векторов для алгоритмов электронной цифровой подписи Вестник Санкт-Петербургского университета, серия 10 вып. 1, 2009, с. 96-102.

Надійшла до редколегії 3.04.2012

Бойко Артем Олександрович, аспірант кафедри БІТ ХНУРЕ. Область наукових інтересів: функції гешування, побудування високошвидкісних систем захисту інформації.

Халімов Геннадій Зайдулович, кандидат технічних наук, доцент кафедри БІТ ХНУРЕ. Область наукових інтересів: методи та засоби автентифікації даних.

УДК 004.056

Метод универсального хеширования по рациональным функциям алгебраических кривых над кольцами / А.А. Бойко, Г.З. Халимов // Прикладная радиоэлектроника: науч.-техн. журнал. – Том 11. № 2. – С. 165–170.

Рассмотрены известные методы универсального хеширования. Описаны их недостатки, в частности наличие уязвимости к наблюдению за временем исполнения. Для решения данной проблемы предложено заменить вычисления в полях на вычисления в кольцах. Предложен метод универсального хеширования по рациональным функциям алгебраических кривых над кольцами векторов, который в отличие от существующих методов универсального хеширования в полях обеспечивает более высокое быстродействие и не является уязвимым к наблюдению времени исполнения.

Ключевые слова: универсальная функция хеширования, переобразования с кольцами.

Табл. 2. Библиогр.: 6 назв.

UDC 004.056

Technique of universal hashing by rational functions of algebraic curves over rings / A.A. Boiko, G.Z. Halimov // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 165–170.

The paper considers the known techniques of universal hashing and describes their shortcomings, in particular, vulnerabilities to observation of implementation time. It is suggested that calculations over fields be replaced with calculations over rings to solve the problem. A technique of universal hashing by rational functions of algebraic curves over vector rings is suggested which unlike the existing ones over fields is faster and invulnerable to the observation of implementation time.

Keywords: universal hashing, transformations over rings.

Tab. 2. Ref.: 6 items.

УДК 681.3.06

ИССЛЕДОВАНИЕ КОЛЛИЗИОННЫХ СВОЙСТВ КОДОВ АУТЕНТИФИКАЦИИ СООБЩЕНИЙ UMAC

А.А. КУЗНЕЦОВ, О.Г. КОРОЛЬ, С.П. ЕВСЕЕВ

Рассматривается алгоритм формирования кодов аутентификации сообщений UMAC, в основе которого лежит использование универсальных хеширующих функций. Предлагается уменьшенная модель UMAC (mini-UMAC) и методика статистического исследования коллизионных свойств формируемых кодов аутентификации сообщений. С использованием уменьшенной модели UMAC исследуются коллизионные свойства кодов аутентификации, показано, что применение криптографического преобразования (с использованием алгоритма AES) на завершающем этапе UMAC приводит к нарушению свойств универсального хеширования.

Ключевые слова: мини-UMAC, аутентификация, универсальное хеширование, коды аутентичности, алгоритм AES.

ПОСТАНОВКА ПРОБЛЕМЫ В ОБЩЕМ ВИДЕ И АНАЛИЗ ЛИТЕРАТУРЫ

Эффективным механизмом обеспечения целостности и аутентичности информации в современных телекоммуникационных системах и сетях является хеширование информации [1 – 7], применяемое как для формирования кодов обнаружения манипуляций (MDC – Manipulation Detection Code), так и для построения кодов аутентификации сообщений (MAC – Message Authentication Code) [5, 7].

Проведенный анализ показал, что наибольшей вычислительной эффективностью обладает отобранный при проведении европейского конкурса NESSIE алгоритм UMAC (Message Authentication Code using Universal Hashing) [5, 7], для формирования кодов аутентификации в котором используются семейства универсальных хеширующих функций [8, 9]. Число коллизий (столкновений) формируемых хеш-образов для каждого введенного ключа универсального хеширования не превышает некоторой заранее заданной величины, а криптостойкость UMAC обеспечивается на уровне выбранного криптоалгоритма (по спецификации рекомендован алгоритм шифрования AES). Однако влияние используемого криптоалгоритма на коллизионные свойства кодов подлинности сообщений UMAC на сегодняшний день не исследовано, обеспечение свойств универсального хеширования в такой многослойной конструкции не обосновано [1 – 7].

Целью данной работы является исследование коллизионных свойств хеширующих функций алгоритма UMAC, оценка влияния применяемого криптографического преобразования на последнем этапе формирования кодов аутентификации на обеспечение свойств универсального хеширования. Для этого в первой части статьи приводится общая конструкция схемы формирования кодов аутентификации сообщений UMAC, исследуются основные этапы (слои) преобразования для построения ключевых итерационных хеширующих функций. Во второй части статьи предлагается уменьшенная модель UMAC (mini-UMAC), позволяющая при сохранении математической структуры основных преобразований за счет уменьшения ключевого пространства и пространства аутентификаторов оценить число возникающих коллизий. Методика статистического исследования коллизионных свойств формируемых кодов аутентификации сообщений, с использованием уменьшенной модели UMAC, приводиться в третьей части статьи. Результаты моделирования и обсуждение полученных данных приводятся в четвертой части статьи, по которым делается вывод о нарушении свойств универсального хеширования UMAC.

1. ФОРМИРОВАНИЕ КОДОВ АУТЕНТИФИКАЦИИ СООБЩЕНИЙ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМА UMAC

Одна из первых версий алгоритма формирования кодов подлинности сообщений с использованием универсального хеширования (UMAC) была представлена в работе [1]. В дальнейшем, после некоторой доработки [2 – 4] алгоритм UMAC был представлен в финальном отчете европейского конкурса NESSIE – New European Schemes for Signatures, Integrity and Encryption (новые европейские схемы для подписей, целостности и шифрования) [5]. Одна из последних электронных версий алгоритма UMAC доступна в электронном виде [6]. Наиболее подробно отдельные компоненты UMAC изложены в диссертационной работе [7].

Рассмотрим общую схему формирования кодов подлинности сообщений с использованием алгоритма UMAC, проанализируем основные аналитические соотношения, описывающие внутреннюю структуру и применяемые преобразования при формировании кодов подлинности сообщений.

1.1. Общая схема формирования кодов подлинности сообщений с использованием алгоритма UMAC. Код подлинности сообщений (обозначим его *Tag*) по спецификации алгоритма

UMAC формируется посредством вычисления следующей функции:

$$Tag = UMAC(K, M, Nonce, Taglen) = Y \oplus Pad,$$

где: K – секретный ключ, длина которого $Keylen$ равна стандартной длине секретного ключа используемого блочного симметричного шифра (спецификацией UMAC рекомендуется использовать алгоритм шифрования AES (FIPS-197), в этом случае длина секретного ключа $Keylen$ принадлежит множеству допустимых значений $\{16, 24, 32\}$ байт); M – информационное сообщение, подлежащее аутентификации, представленное в виде массива-строки размерностью от одного до 2^{67} бит (2^{64} байт); $Nonce$ – неповторяющееся (для всех вводимых информационных сообщений M) восьмибайтное число; $Taglen$ – целое число из множества допустимых значений $\{4, 8, 12, 16\}$, задающее длину кода подлинности сообщений Tag в байтах; $Hash(K, M, Taglen)$ – функция ключевого универсального хеширования информационного сообщения M с использованием секретного ключа K; $PDF(K, Nonce, Taglen)$ – функция формирования псевдослучайной подложки (Pad) по введенному значению $Nonce$ и секретному ключу K; «$\oplus$» – побитовое сложение (XOR) результата ключевого хеширования сообщения $Y = Hash(K, M, Taglen)$ и сформированной подложки

$$Pad = PDF(K, Nonce, Taglen), \text{ т.е.}$$

$$Tag = Hash(K, M, Taglen) \oplus PDF(K, Nonce, Taglen).$$

Длина хеш-кода Y, подложки Pad и кода Tag принадлежат множеству допустимых значений $\{32, 64, 96, 128\}$ бит. Эти фиксированные значения $Taglen$ соответствуют случаю формирования кодов подлинности сообщений UMAC – 32, UMAC – 64, UMAC – 96 или UMAC – 128, соответственно.

Рассмотрим схему формирования хеш-кодов $Y = Hash(K, M, Taglen)$ и подложки $Pad = PDF(K, Nonce)$.

1.2. Схема формирования хеш-кодов $Y = Hash(K, M, Taglen)$. Вычисление значения функции $Hash(K, M, Taglen)$ ключевого универсального хеширования информационного сообщения M с использованием секретного ключа K выполняется в три этапа (используется три уровня (слоя) ключевого хеширования) $Hash_{L1}$, $Hash_{L2}$ и $Hash_{L3}$, соответственно. Второй уровень хеширования $Hash_{L2}$ выполняется только если длина хешируемого сообщения M превосходит 1024 байт.

Длина хеш-кода Y кратна 32 битам, его значение $Y = Hash(K, M, Taglen)$ для любой длины $Taglen$ формируется посредством объединения (конкатенации) нескольких (от одной до четырех) последовательностей Y_{L3_i}

$$Y = Hash(K, M, Taglen) = Y_{L3_1} \| Y_{L3_2} \| ... \| Y_{L3_{It}},$$

$$It = Taglen / 4,$$

где Y_{L3_i} – результат многоуровневого хеширования сообщения M на i-ой итерации с использованием соответствующих ключей, $i = 1, 2, ..., It$.

Рассмотрим формирование хеш-кода Y_{L3_i} на i-ой итерации. Для этого обозначим результат многоуровневого хеширования на произвольной i-ой итерации следующим образом:

$$Y_{L3_i} = Y_{L3} =$$

$$= Hash_{L3}\left(K_{L3_1}, K_{L3_2}, Hash_{L2}\left(K_{L2}, Hash_{L1}(K_{L1}, M)\right)\right),$$

где: $Hash_{L1}(K_{L1}, M)$, $Hash_{L2}(K_{L2}, Y_{L1})$ и $Hash_{L3}\left(K_{L3_1}, K_{L3_2}, Y_{L2}\right)$ – функции ключевого хеширования первого, второго и третьего уровня, управляемые и зависящими от номера итерации секретными ключами K_{L1}, K_{L2}, K_{L3_1}, K_{L3_2}, соответственно.

Ключевые последовательности K_{L1}, K_{L2}, K_{L3_1}, K_{L3_2} формируются по введенному секретному ключу K длины $Keylen$ байт с использованием специальной функции $KDF(K, Index, Numbyte)$ (Key-Derivation Function – KDF), где $Index$ и $Numbyte$ представляют собой два целых положительных числа, не превосходящих 2^{64}.

Первый уровень хеширования выполняет разбиение массива-строки M размерности до 2^{64} байт на блоки M_i по 1024 байт с последующим преобразованием каждого блока функцией $NH(K_{L1}, M_i)$. Полученные результаты $Hash_{L1_i} = NH(K_{L1}, M_i)$ конкатенируются (объединяются) в строку $Y_{L1} = Hash_{L1}(K_{L1}, M)$, которая короче информационной последовательности в 128 раз. Эта строка и является результатом хеширования первого уровня:

$$Y_{L1} = Hash_{L1}(K_{L1}, M) =$$

$$= NH(K_{L1}, M_0) \| NH(K_{L1}, M_1) \| ... \| NH(K_{L1}, M_{n-1}),$$

где $n = \left\lceil \frac{Length(M)}{1024} \right\rceil$, $\lceil x \rceil$ – целая часть числа x, $Length(M)$ – байтовая длина информационного сообщения M.

Значение функции $Hash_{L1_i} = NH(K_{L1}, M_i)$ вычисляется по следующему правилу. Информационный блок M_i разбивается на четырехбайтовые подблоки так, что

$$M_i = M_{i_1} \| M_{i_2} \| ... \| M_{i_t},$$

где $t = \left\lceil \frac{Length(M_i)}{4} \right\rceil$. В данном случае

$$t = \left\lceil \frac{1024}{4} \right\rceil = 256.$$

Аналогичным образом ключевая последовательность K_{L1} представляется в виде последовательностей четырехбайтовых подблоков:

$$K_{L1} = K_{L1_1} \| K_{L1_2} \| ... \| K_{L1_t}.$$

После чего (принимая начальное состояние $Hash_{L1_i} = 0$) для всех $j = 1,9,17...,t-7$ выполняются следующие операции:

$$Hash_{L1_i} = Hash_{L1_i} +$$
$$+_{64}((M_{i_{j+0}} +_{32} K_{L1_{j+0}}) \times_{64} (M_{i_{j+4}} +_{32} K_{L1_{j+4}})),$$
$$Hash_{L1_i} = Hash_{L1_i} +$$
$$+_{64}((M_{i_{j+1}} +_{32} K_{L1_{j+1}}) \times_{64} (M_{i_{j+5}} +_{32} K_{L1_{j+5}})),$$
$$Hash_{L1_i} = Hash_{L1_i} +$$
$$+_{64}((M_{i_{j+2}} +_{32} K_{L1_{j+2}}) \times_{64} (M_{i_{j+6}} +_{32} K_{L1_{j+6}})),$$
$$Hash_{L1_i} = Hash_{L1_i} +$$
$$+_{64}((M_{i_{j+3}} +_{32} K_{L1_{j+3}}) \times_{64} (M_{i_{j+7}} +_{32} K_{L1_{j+7}})),$$

где $+_{64}$, $+_{32}$ – операции сложения по модулю 2^{64} и 2^{32}, соответственно; $\times_{64}$ – операция умножения по модулю 2^{64}.

В работах [1 – 7] показано, что рассмотренная функция ключевого хеширования *NH* принадлежит к классу универсальных хеширующих функций.

Второй уровень хеширования использует полиномиальное ключевое хеширование *Poly*, подробно рассмотренное в работах [1 – 7]. Результатом работы этого уровня есть вычисление хеш-кода

$$Y_{L2} = Hash_{L2}(K_{L2}, Y_{L1}) =$$
$$= Poly(Wordbits, Maxwordrange, k, M_P),$$

т.е. на вход хеширования второго уровня подается строка $Y_{L1} = Hash_{L1}(K_{L1}, M)$.

В качестве исходных данных функция полиномиального хеширования использует:

$Wordbits \in [64,128]$; *Maxwordrange* – положительное целое число, меньшее $2^{Wordbits}$; k – зависящее от ключа K_{L2} целое число из диапазона $[0,..., prime(Wordbits)-1]$, $prime(x)$ – наибольшее простое число, меньшее 2^x;

$M_P = Y_{L1} = Hash_{L1}(K_{L1}, M)$ – данные, подлежащие полиномиальному хешированию.

По спецификации алгоритма UMAC в качестве $prime(x)$ используются следующие константы: $prime(36) = 2^{36} - 5$, $prime(64) = 2^{64} - 59$, $prime(128) = 2^{128} - 159$. Битовую длину M_P обозначим $Bytelength(M_P)$. В зависимости от длины M_P используются следующие особенности в реализации второго уровня хеширования:

– если длина поступивших данных M_P не превосходит 2^{17} байт, тогда полиномиальное хеширование *Poly* выполняется с параметрами $Wordbits = 64$; $Maxwordrange = 2^{64} - 2^{32}$; $k = k64$ – строка, образованная первыми восемью байтами ключа K_{L2} и специальной восьмибайтной маской;

– если длина поступивших данных M_P превосходит 2^{17} байт (но не превосходит 2^{64} байт), тогда первые 2^{17} байт данных обрабатываются функцией полиномиального хеширования $Poly(64, 2^{64} - 2^{32}, k64, M_P)$, а оставшиеся байты данных обрабатываются функцией *Poly* с параметрами $Wordbits = 128$; $Maxwordrange = 2^{128} - 2^{96}$; $k = k128$ – строка, образованная последними 16 байтами ключа K_{L2} и специальной 16 байтной маской.

Хешируемые данные M_P разбиваются на блоки по $Wordbytes = Wordbits / 8$ байт:

$$M_P = M_{P_1} \| M_{P_2} \| ... \| M_{P_n},$$

где $n = Bytelength(M_P) / Wordbytes$.

Результатом хеширования является значение полиномиальной функции

$$Y_{L2} = \left(M_{P_n} + kM_{P_{n-1}} + ... + k^{n-1} M_{P_1} + k^n\right) \bmod(p),$$

которое вычисляется итеративной процедурой (для всех $i = 1,2,...,n$):

$$Poly_i = \left(kPoly_{i-1} + M_{P_i}\right) \bmod(p), \quad Poly_0 = 1,$$
$$p = prime(Wordbits)$$

с помощью схемы Горнера

$$M_{P_n} + kM_{P_{n-1}} + ... + k^{n-1} M_{P_1} + k^n =$$
$$= (((k + M_{P_1})k + M_{P_2})k + ... + M_{P_{n-1}})k + M_{P_n}.$$

Вычисленное хеш-значение $Y_{L2} = Poly_n$ является целым числом из диапазона

$$[0,..., prime(Wordbits) - 1].$$

Рассмотренная функция полиномиального ключевого хеширования

$$Poly(Wordbits, Maxwordrange, k, M_P)$$

принадлежит к классу универсальных хеширующих функций [1 – 7].

Третий уровень хеширования

$$Hash_{L3}\left(K_{L3_1}, K_{L3_2}, Y_{L2}\right)$$

выполняется над результатом полиномиального хеширования и преобразует поданные на его вход данные длины до 16 байт в хеш-код Y фиксированной длины 32 бита.

В качестве исходных данных третьего уровня хеширования выступают две ключевых последовательности K_{L3_1} и K_{L3_2} длины 64 и 4 байта соответственно, а также входная 16 байтная последовательность Y_{L2}.

Хешируемые данные Y_{L2} и ключевая последовательность K_{L3_1} равномерно разбиваются на восемь блоков, каждый из которых представляется как целое число Y_{L2_i} и $K_{L3_{1i}}$, $i = 1,2,...,8$.

Хеш-значение Y_{L3} вычисляется следующим образом:

$$Y_{L3} =$$
$$= \left(\left(\left(\sum_{i=1}^{m} Y_{L2_i} K_{L3_{1i}}\right) \bmod(prime(36))\right) \bmod(2^{32})\right) xor(K_{L3_2}),$$

где $(x)xor(y)$ – операция «исключающего ИЛИ» над значениями x и y.

Рассмотренная функция ключевого хеширования $Y_{L3} = Hash_{L3}\left(K_{L3_1}, K_{L3_2}, Y_{L2}\right)$ принадлежит к

классу универсальных хеширующих функций, ее свойства подробно исследованы в работах [1–7].

1.3. Схема формирования ключей (KDF: Key-Derivation Function). Специальная функция $KDF(K, Index, Numbyte)$ предназначена для формирования последовательностей псевдослучайных бит данных, которые используются на различных уровнях формирования кодов подлинности сообщений как ключевые данные соответствующих функций хеширования.

В качестве исходных данных функции генерации ключевых псевдослучайных последовательностей используется секретных ключ K длины $Keylen$ байт и два положительных целых числа $Index$ и $Numbyte$, значение которых не превосходит 2^{64}.

Для формирования псевдослучайных ключевых последовательностей используется блочный симметричный шифр. Обозначим процедуру шифрования блока данных T длины $Blocklen$ байт с использованием секретного ключа K длины $Keylen$ байт в виде некоторой функции $Enchiper(K,T)$. Тогда процедуру формирования псевдослучайной ключевой последовательности $K' = KDF(K, Index, Numbyte)$ можно представить в виде следующего итеративного (для всех $i = 1,2,...,n$) преобразования:

$$T_i = Index \| i,$$

$$K'_i = Enchiper(K, T_i),$$

$$K' = K'_1 \| K'_2 \| ... \| K'_n,$$

где $n = \left\lceil \frac{Numbyte}{Blocklen} \right\rceil$, $\lceil x \rceil$ – целая часть числа x, $a \| b$ – конкатенация (присоединение) строк a и b.

Сформированная последовательность псевдослучайных ключевых бит данных K' имеет длину $Numbyte$ байт, кратную длине блока $Blocklen$ байт.

1.4. Схема формирования псевдослучайной подложки (PDF: Pad-Derivation Function). Функция $PDF(K, Nonce, Taglen)$ предназначена для формирования псевдослучайной подложки Pad, используемой на заключительном этапе формирования кода подлинности сообщения.

В качестве исходных данных используется секретных ключ K длины $Keylen$ байт и неповторяющееся (для всех вводимых информационных сообщений M) восьмибайтное число $Nonce$, а также целое число $Taglen$, задающее размер (длину в байтах) формируемого кода подлинности Tag.

Процедура формирования псевдослучайной подложки $Pad = PDF(K, Nonce, Taglen)$ состоит в формировании подключа

$$K' = KDF(K, Index, Numbyte),\ Index = 0,$$

$$Numbyte = Keylen,$$

с использованием рассмотренной выше процедуры формирования последовательностей псевдослучайных ключевых бит и шифрования значения $Nonce$ на сформированном подключе K', т.е.:

$$Pad = PDF(K, Nonce, Taglen) =$$

$$= Enchiper(KDF(K, 0, Keylen), Nonce).$$

Процедура формирования псевдослучайной подложки Pad построена так, что результирующее значение Pad имеет длину $Taglen$ байт вне зависимости от значений $Blocklen$ и $Nonce$.

Таким образом, рассмотренная схема формирования кодов подлинности сообщений UMAC использует многоуровневую конструкцию универсального хеширования $Hash(K, M, Taglen)$ и процедуру формирования псевдослучайной подложки Pad. Применение универсального хеширования позволяет обеспечить равновероятность формирования хеш-образов для всего множества используемых ключевых данных, на чем и базируется доказательство безопасности алгоритма [1–7]. Формирование псевдослучайной подложки криптографически стойким алгоритмом (например, с использованием блочного симметричного шифра AES) обеспечивает криптостойкость алгоритма UMAC на уровне стойкости применяемого криптоалгоритма [5, 7]. Следовательно, рассмотренная схема формирования UMAC обладает потенциально высокими показателями эффективности.

В тоже время на сегодняшний день не исследованы коллизионные свойства алгоритма UMAC после применения завершающей процедуры наложения на формируемые хеш-коды $Y = Hash(K, M, Taglen)$ псевдослучайных подложек $Pad = PDF(K, Nonce, Taglen)$. Ниже показано, что результирующие коды подлинности сообщений $Tag = UMAC(K, M, Nonce, Taglen) = Y \oplus Pad$ формируются не равновероятно для всего множества используемых ключевых данных. Следовательно алгоритм формирования UMAC после применения последнего слоя наложения псевдослучайных подложек теряет свойство «универсальности» хеширования его коллизионные свойства существенно ухудшаются.

Для проведения исследований коллизионных свойств кодов аутентификации сообщений, сформированных по рассмотренной выше схеме, предлагается использовать уменьшенную модель UMAC (mini-UMAC). Применение уменьшенных моделей позволяет, сохранив алгебраическую структуру криптоалгоритма, исследовать основные показатели его эффективности [10 – 14]. Этот поход широко используется на сегодняшний день при исследовании криптографических свойств блочных симметричных шифров. Так, например, в работах [12 – 14] разработаны уменьшенные модели криптоалгоритмов AES, Camelia, ADE, Лабиринт, Калина, Мухомор и др., использование которых позволило экспериментально исследовать дифференциальные и

линейные свойства соответствующих шифров, оценить их устойчивость к атакам дифференциального и линейного криптоанализа. Коме того, на основе анализа уменьшенных моделей в работах [12 – 14] предложен подход к оценке эффективности блочных симметричных шифров в виде вычислительных затрат, требуемых для достижения шифром асимптотических характеристик случайной подстановки.

В настоящей работе предлагается дальнейшее развитие данного направления, состоящее в использовании уменьшенных моделей отдельных слоев преобразований для оценки коллизионных свойств, формируемых кодов аутентификации сообщений.

2. УМЕНЬШЕННАЯ МОДЕЛЬ UMAC (MINI-UMAC)

Схема формирования кодов аутентификации сообщений UMAC использует в своей структуре несколько слоев преобразования, в том числе блочный симметричных шифр (рекомендован к использованию шифр AES). Разрабатываемая уменьшенная модель UMAC должна включать соответствующие слои преобразования с сохранением их алгебраической структуры при выполнении масштабирования до мини-версии. Естественным представляется исследовать коллизионные характеристики формируемых образов (кодов) на каждом из слоев преобразования, в том числе формируемых с помощью блочного симметричного шифра псевдослучайных подложек Pad, проанализировать их влияние на коллизионные свойства в целом, т.е. на коллизионные свойства кодов аутентификации сообщений уменьшенной модели UMAC.

Выше было показано, что схема формирования кодов UMAC состоит из следующих слоев:

– трехуровневое универсальное хеширование для формирования хеш-кодов

$$Y = Hash(K, M, Taglen);$$

– криптографическое преобразования с использованием блочного симметричного шифра для формирования псевдослучайной подложки

$$Pad = PDF(K, Nonce, Taglen);$$

– заключительное преобразование для формирования кодов аутентификации сообщений

$$Tag = UMAC(K, M, Nonce, Taglen) = Y \oplus Pad .$$

Рассмотрим каждый слой схемы формирования кодов аутентификации сообщений UMAC на предмет их масштабирования.

2.1. Мини-версию трехуровневого универсального хеширования построим без изменения структуры алгебраических преобразований простым уменьшением размерности блоков обрабатываемых данных в восемь раз.

Соответствующая длина хеш-кода $Y_{\min i}$ уменьшенной модели первого слоя будет кратна 4 битам, его значение сформируем посредством объединения (конкатенации) четырех последовательностей $Y_{\min iL3_i}$

$$Y_{\min i} = Y_{\min iL3_1} \| Y_{\min iL3_2} \| Y_{\min iL3_3} \| Y_{\min iL3_4} ,$$

где $Y_{\min iL3_i}$ – результат многоуровневого хеширования сообщения уменьшенной модели первого слоя mini-UMAC.

Рассмотрим процесс формирования одного блока $Y_{\min iL3_i}$ (второй уровень хеширования в уменьшенной модели выполнять не будем):

$$Y_{\min iL3_i} = Y_{\min iL3} =$$

$$= Hash_{\min iL3}\left(K_{\min iL3_1}, K_{\min iL3_2}, Hash_{\min iL1}\left(K_{\min iL1}, M_{\min i}\right)\right),$$

где $K_{\min iL1}$, $K_{\min iL3_1}$, $K_{\min iL3_2}$ – ключевые последовательности mini-UMAC, $Hash_{\min iL1}$ и $Hash_{\min iL3}$ – уменьшенные версии хеширования первого и третьего уровней соответственно.

На первом уровне массив-строка $M_{\min i}$ размерности 32 бита преобразуется функцией $NH(K_{L1}, M_i)$. Эта строка и является результатом хеширования первого уровня: $Y_{\min iL1} = NH_{\min i}(K_{\min iL1}, M_{\min i})$.

Значение функции $NH_{\min i}(K_{\min iL1}, M_{\min i})$ вычисляется по следующему правилу. Информационный блок $M_{\min i}$ разбивается на восемь четырехбитовых подблоков

$$M_{\min i} = M_{\min i_1} \| M_{\min i_2} \| ... \| M_{\min i_8} .$$

Аналогичным образом ключевая последовательность K_{L1} представляется в виде последовательностей из восьми четырехбитовых подблоков: $K_{\min iL1} = K_{\min iL1_1} \| K_{\min iL1_2} \| ... \| K_{\min iL1_8}$.

После чего (принимая начальное состояние $Hash_{L1} = 0$) выполняются следующие операции:

$$Hash_{\min iL1} = Hash_{\min iL1} +$$
$$+_8((M_{\min i_0} +_4 K_{\min iL1_0}) \times_8 (M_{\min i_4} +_4 K_{\min iL1_4})),$$

$$Hash_{\min iL1} = Hash_{\min iL1} +$$
$$+_8((M_{\min i_1} +_4 K_{\min iL1_1}) \times_8 (M_{\min i_5} +_4 K_{\min iL1_5})),$$

$$Hash_{\min iL1} = Hash_{\min iL1} +$$
$$+_8((M_{\min i_2} +_4 K_{\min iL1_2}) \times_8 (M_{\min i_6} +_4 K_{\min iL1_6})),$$

$$Hash_{\min iL1} = Hash_{\min iL1} +$$
$$+_8((M_{\min i_3} +_4 K_{\min iL1_3}) \times_8 (M_{\min i_7} +_4 K_{\min iL1_7})),$$

где $+_8$, $+_4$ – операции сложения по модулю 2^8 и 2^4, соответственно; $\times_8$ – операция умножения по модулю 2^8.

В результате вычислений формируется восьмибитное значение $Y_{\min iL1} = Hash_{\min iL1}$.

Третий уровень хеширования преобразует поданные на его вход восьмибитные данные $Y_{\min iL1}$ в хеш-код $Y_{\min iL3}$ длины 4 бита. В качестве ключевых последовательностей выступают $K_{\min iL3_1}$ и $K_{\min iL3_2}$ длины 16 и 4 бита соответственно.

Хешируемые данные $Hash_{\min iL1}$ и ключевая последовательность $K_{\min iL3_1}$ равномерно разбиваются на четыре блока, каждый из которых представляется как целое число $Y_{\min iL2_i}$ и $K_{\min iL3_{1i}}$, $i=1,2,...,4$.

Хеш-значение $Y_{\min iL3}$ вычисляется следующим образом:

$$Y_{\min iL3} =$$

$$=\left(\left(\left(\sum_{i=1}^{4} Y_{\min iL2_i} K_{\min iL3_{1i}}\right)\bmod(17)\right)\bmod(2^4)\right) xor(K_{\min iL3_2}),$$

где $(x)xor(y)$ – операция «исключающего ИЛИ» над значениями x и y.

2.2. Мини-версия блочного симметричного шифра AES для формирования псевдослучайной подложки подробно рассмотрена в работах [10-14]. Наиболее простой в реализации есть мини-версия шифра AES (Baby-Rijndael), которая предложена К. Бергманом [10]. Кратко рассмотрим эту уменьшенную модель шифра и обоснуем ее использование для формирования псевдослучайной подложки в mini-UMAC.

Размер блока открытого текста равен 16 бит, которые обозначим четырьмя шестнадцатеричными числами h_0, h_1, h_2, h_3. Отметим, что h_0 состоит из первых четырех бит входного потока. Однако когда h_0 рассматривается как шестнадцатеричная цифра, первый бит рассматривается, как бит высшего порядка. Например, входной блок 1000 1100 0111 0001 будет представлен $h_0=8$, $h_1=c$, $h_2=7$, $h_3=1$.

Размер ключа также равен 16 бит. Обозначим его как 4 шестнадцатеричных чисел k_0, k_1, k_2, k_3.

Шаги шифра применяются к состоянию – массиву 2×2 шестнадцатеричных цифр. Однако для рассматриваемой ниже операции $\tilde{\sigma}$ состояние будет представлено как массив 8×2 бит, т.е. каждая шестнадцатеричная цифра будет, рассматривается как столбец 4 бит с битом высшего порядка сверху.

Входной блок загружается в состояние отображениемv h_0, h_1, h_2, h_3 в $\begin{bmatrix} h_0 & h_2 \\ h_1 & h_3 \end{bmatrix}$. Например, входной блок 1000 1100 0111 0001 будет загружен как $\begin{bmatrix} 8 & 7 \\ c & 1 \end{bmatrix}$, где матрица 8×2 будет $\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix}$.

Baby-Rijndael включает несколько идентичных по структуре раундов (по умолчанию их 4). Перед шифрованием входной блок загружается в состояние, как описано выше и рассчитываются раундовые ключи. Шифрование имеет общую структуру:

$$E(a)=r_4\circ r_3\circ r_2\circ r_1\circ(a\oplus k_0),$$

где a обозначает состояние, k_0, k_1, k_2, k_3, k_4 – раундовые ключи и $r_i(a)=(t\cdot\tilde{\sigma}(S(a)))\oplus k_i$, за исключением r_4, где пропущено умножение на t. В конце шифра состояние сгружается в 16-битный блок в таком же порядке, в котором он загружался.

Теперь опишем отдельные компоненты шифра.

SubBytes: Операция S есть выборочная таблица, которая применяется к каждой 16-ричной цифре состояния:

$$\begin{bmatrix} h_0 & h_2 \\ h_1 & h_3 \end{bmatrix} \xrightarrow{S} \begin{bmatrix} S(h_0) & S(h_2) \\ S(h_1) & S(h_3) \end{bmatrix},$$

где функция S задается следующей таблицей 1.

Таблица 1

Выборочная таблица, реализующая S-блок Baby-Rijndael

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	a	4	3	b	8	e	2	c	5	7	6	f	0	1	9	d

ShiftRows: Операция $\tilde{\sigma}$ просто меняет входы во второй строке состояния:

$$\begin{bmatrix} h_0 & h_2 \\ h_1 & h_3 \end{bmatrix} \xrightarrow{\tilde{\sigma}} \begin{bmatrix} h_0 & h_2 \\ h_3 & h_1 \end{bmatrix}.$$

MixColumns: Матрица t является следующей 8Ч8 матрицей бит:

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Для этого преобразования состояние рассматривается как 8×2 битовая матрица. Состояние умножается слева на t, используя матричное умножение по модулю 2: $a=ta$.

KeySchedule: В начале шифра и в конце каждого раунда состояние побитно складывается (т.е. по модулю 2) с раундовым ключом. Столбцы раундовых ключей определены рекурсивно следующим образом:

$$w_0=\begin{pmatrix} k_0 \\ k_1 \end{pmatrix},\ w_1=\begin{pmatrix} k_2 \\ k_3 \end{pmatrix},$$

$$w_{2i}=w_{2i-2}\oplus S(reverse(w_{2i-2}))\oplus r_i,$$

$$\{w_{2i+1}=w_{2i-1}\oplus w_{2i}\}$$

для всех $i=1,2,3,4$, где $r_i=\begin{pmatrix} 2^{i-1} \\ 0 \end{pmatrix}$, а функция *reverse* заменяет два входа в столбец. Функция S та же, что и описанная выше.

Следует заметить, что все сложения выполняются побитно по модулю 2. Наконец, для

$i = 1,2,3,4$ раундовый ключ k_i есть матрица, чьи столбцы есть w_{2i} и w_{2i+1}.

Использование рассмотренной уменьшенной модели блочного симметричного шифра AES позволяет провести экспериментальные исследования коллизионных свойств формируемых псевдослучайных подложек по всему множеству секретных ключей. Так, псевдослучайная подложка $Pad_{\min i}$ mini-UMAC формируется посредством шифрования неповторяющегося для каждого информационного сообщения $M_{\min i}$ числа *Nonce*. Результирующее значение $Pad_{\min i}$ имеет длину 16 бит, так же, как и соответствующая длина хеш-кода $Y_{\min i}$.

2.3. Мини-версия заключительного преобразования для формирования кодов аутентификации сообщений mini-UMAC состоит в поразрядном суммировании по модулю 2 значений $Y_{\min i}$ и $Pad_{\min i}$: $Tag_{\min i} = Y_{\min i} \oplus Pad_{\min i}$.

Таким образом, масштабирование применяемых преобразований на соответствующих слоях схемы формирования кодов аутентификации сообщений, позволяет построить уменьшенную модель UMAC, экспериментально исследовать коллизионные свойства формируемых образов (кодов). Коэффициент масштабирования при разработке мини-модели UMAC выбран таким образом, чтобы длина формируемых хеш-кодов Y, псевдослучайных подложек Pad и кодов аутентификации сообщений $Tag = Y \oplus Pad$ была равна длине блока мини-версии блочного симметричного шифра AES [10], т.е. 16 битам. Выбор такого коэффициента масштабирования позволяет с одной стороны сохранить алгебраическую структуру основных преобразований алгоритма UMAC, в том числе и входящего в его схему алгоритма AES, с другой стороны это дает возможность провести экспериментальные исследования с использованием методов статистической проверки гипотез и математической статистики, рассматривая ограниченный набор элементов Y, Pad и $Tag = Y \oplus Pad$ и соответствующие результаты по оценке числа коллизий как выборку из генеральной совокупности.

Обоснуем методику статистического исследования коллизионных свойств формируемых элементов (обозначим их для простоты $h(x)$), рассмотрим основные условия и ограничения при проведении экспериментов.

3. МЕТОДИКА СТАТИСТИЧЕСКОГО ИССЛЕДОВАНИЯ КОЛЛИЗИОННЫХ СВОЙСТВ

Проведение экспериментальных исследований коллизионных свойств кодов аутентификации сообщений UMAC проведем по соответствующим слоям преобразования:

1. На первом этапе исследуем коллизионные свойства мини-версии универсального хеширования. Для этого необходимо подтвердить в ходе эксперимента теоретические оценки числа возникающих коллизий формируемых хеш-кодов $Y_{\min i}$;

2. На втором этапе проведем экспериментальные исследования коллизионных свойств псевдослучайных подложек $Pad_{\min i}$ на основе анализа свойств уменьшенной модели шифра Baby-Rijndael. Подобные исследования в доступной литературе не описаны и, по всей видимости, проводятся нами впервые;

3. На третьем этапе проведем экспериментальные исследования коллизионных свойств формируемых с использованием mini-UMAC кодов аутентификации сообщений $Tag_{\min i} = Y_{\min i} \oplus Pad_{\min i}$. Это наиболее важная часть проводимых исследований, поскольку она позволит ответить на вопрос о сохранении свойств универсального хеширования после применения слоя криптографического преобразования информации.

Оценку числа коллизий формируемых элементов будем проводить, ориентируясь на коллизионные свойства универсального хеширования. Собственно говоря, нам требуется подтвердить или опровергнуть гипотезу о сохранении коллизионных свойств универсального хеширования на всех этапах формирования кодов аутентификации сообщений mini-UMAC.

Идея универсального хеширования заключается в определении такого набора элементов конечного множества H хеш-функций $h: A \to B$, $|A| = a$, $|B| = b$ чтобы случайный выбор функции $h \in H$ обеспечивал бы низкую вероятность коллизии, т.е. для любых различных входов x_1 и x_2 вероятность того, что $h(x_1) = h(x_2)$ (вероятность коллизии, столкновения) не должна превосходить некоторой заранее заданной величины ε:

$$P_{\text{кол}} = P(h(x_1) = h(x_2)) \le \varepsilon,$$

причем вероятность коллизии может быть рассчитана как

$$P_{\text{кол}} = \frac{\delta_H(x_1, x_2)}{|H|},$$

где $\delta_H(x_1, x_2)$ – количество таких хеш-функций в H, при которых значения $x_1, x_2 \in A$, $x_1 \ne x_2$ вызывают коллизию, т.е. $h(x_1) = h(x_2)$.

Приведем два определения универсального хеширования [8, 9].

1. Пусть $0 < \varepsilon < 1$. H является ε – *универсальным хеш-классом (сокращенно* $\varepsilon - U(H, A, B)$*),* если для двух различных элементов $x_1, x_2 \in A$ существует не больше, чем $|H| \cdot \varepsilon$ функций $f \in H$ таких, что $h(x_1) = h(x_2)$, если $\delta_H(x_1, x_2) \le \varepsilon |H|$ для всех $x_1, x_2 \in A$, $x_1 \ne x_2$.

2. Пусть $0 < \varepsilon < 1$. H является ε – *строго универсальным хеш-классом* (сокращенно $\varepsilon - SU(H, A, B)$) если выполняются следующие условия:

– для каждого $x_1 \in A$ и для каждого $y_1 \in B$,

$$\left|\{h \in H : h(x_1) = y_1\}\right| = |H| / |B|;$$

– для каждого $x_1, x_2 \in A$, $x_1 \neq x_2$ и для каждого $y_1, y_2 \in B$,

$$\left|\{h \in H : h(x_1) = y_1, h(x_2) = y_2\}\right| \leq \varepsilon |H| .$$

Определение универсального класса хеш-функций эквивалентно определению такого алгоритма формирования кода аутентификации, при котором число различных правил формирования кода аутентификации (число ключей), при которых существует коллизия (совпадение кодов аутентификации) для двух произвольных входных последовательностей, ограничено. Число таких ключей не может превосходить значение $P_{\text{кол}} \cdot |H|$, где $P_{\text{кол}}$ – вероятность коллизии, $|H|$ – число всех правил (ключей).

Определение строго универсального класса хеш-функций эквивалентно определению такого алгоритма формирования кодов аутентификации, при котором будут выполняться следующие условия:

1. Число правил формирования кода аутентификации (число ключей), при которых для произвольной входной последовательности значение кода аутентификации не изменяется, ограничено. Число таких ключей не может превосходить значения $|H|/|B|$, где $|H|$ – число всех ключей, $|B|$ – число возможных состояний кода аутентификации;

2. Число правил формирования кода аутентификации (число ключей), при которых для двух произвольных входных последовательностей соответствующие им значения кода аутентификации не изменяются, ограничено. Число таких ключей не может превосходить значения $P_{\text{кол}}|H|$, где $P_{\text{кол}}$ – вероятность коллизии, $|H|$ – число всех ключей.

Вероятность коллизии кодов аутентификации в схеме со строго универсальным хешированием определяется как $P_{\text{кол}} \leq \varepsilon$.

В основе предлагаемой методики статистического исследования коллизионных свойств формируемых элементов $h(x)$ лежит эмпирическая оценка максимумов числа ключей (правил хеширования) при которых:

1. Для произвольных $x_1, x_2 \in A$, $x_1 \neq x_2$ выполняется равенство

$$h(x_1) = h(x_2) ; \quad (1)$$

2. Для произвольных $x_1 \in A$ и $y_1 \in B$ выполняется равенство

$$h(x_1) = y_1 ; \quad (2)$$

3. Для произвольных $x_1, x_2 \in A$, $x_1 \neq x_2$ и $y_1, y_2 \in B$ выполняются равенства

$$h(x_1) = y_1, h(x_2) = y_2 . \quad (3)$$

Оценка по первому критерию соответствует проверке выполнимости условия для универсального класса хеш-функций, оценка по второму и третьему критерию – условий для строго универсального класса хеш-функций.

Введем следующие обозначения:

$$n_1(x_1, x_2) = \left|\{h \in H : h(x_1) = h(x_2)\}\right|,$$
$$x_1, x_2 \in A, \; x_1 \neq x_2 ;$$
$$n_2(x_1, y_1) = \left|\{h \in H : h(x_1) = y_1\}\right|,$$
$$x_1 \in A, \; y_1 \in B ;$$
$$n_3(x_1, x_2, y_1, y_2) = \left|\{h \in H : h(x_1) = y_1, h(x_2) = y_2\}\right|,$$
$$x_1, x_2 \in A, \; x_1 \neq x_2, \; y_1, y_2 \in B .$$

Первый показатель $n_1(x_1, x_2)$ характеризует число правил хеширования, при которых для заданных $x_1, x_2 \in A$, $x_1 \neq x_2$ выполняется равенство (1), т.е. число ключей, при которых существует коллизия (совпадение хеш-кодов) для двух входных последовательностей x_1 и x_2.

Второй показатель $n_2(x_1, y_1)$ характеризует число правил хеширования, при которых для заданных $x_1 \in A$, $y_1 \in B$ выполняется равенство (2), т.е. число ключей, при которых для входной последовательности x_1 значение хеш-кода y_1 не изменяется.

Третий показатель $n_3(x_1, x_2, y_1, y_2)$ характеризует число правил хеширования, при которых для заданных $x_1, x_2 \in A$, $x_1 \neq x_2$, $y_1, y_2 \in B$ выполняется равенство (3), т.е. число ключей, при которых для двух входных последовательностей x_1 и x_2 соответствующие им значения хеш-кодов y_1 и y_2 не изменяются.

Поскольку число ключей, при которых могут выполняться равенства (1), (2) и (3), не должно превосходить соответствующих им значений $P_{\text{кол}} \cdot |H|$, $|H|/|B|$ и $P_{\text{кол}}|H|/|B|$ проведем оценку максимального числа таких ключей для каждого из рассматриваемого набора элементов.

Ограничимся изучением статистических характеристик максимумов этих величин, а затем сравним полученные результаты с числом $P_{\text{кол}} \cdot H$ (для первого критерия), с числом $|H|/|B|$ (для второго критерия) и числом $P_{\text{кол}} \cdot H$ (для третьего критерия).

Таким образом, в качестве статистических показателей оценки коллизионных свойств, по которым будем проводить экспериментальные исследования, предлагается использовать:

– математические ожидания $m(n_1)$, $m(n_2)$ и $m(n_3)$ максимумов числа правил хеширования, при которых выполняются равенства (1), (2) и (3), соответственно;

– дисперсии $D(n_1)$, $D(n_2)$ и $D(n_3)$, характеризующие рассеивание значений числа правил хеширования, при которых выполняются равенства (1), (2) и (3), относительно их математических ожиданий $m(n_1)$, $m(n_2)$ и $m(n_3)$, соответственно.

Оценку коллизионных свойств по приведенным критериям будем производить в среднестатистическом смысле. Другими словами, при постановке эксперимента будем использовать ограниченный набор элементов $x_1, x_2 \in A$, $x_1 \neq x_2$

и соответствующих им хеш-образов $y_1, y_2 \in B$, рассматривая соответствующие результаты как выборку из генеральной совокупности.

Естественной оценкой для математического ожидания m случайной величины X является среднее арифметическое ее наблюдаемых значений X_i (или статистическое среднее) [15]

$$\tilde{m} = \frac{1}{N}\sum_{i=1}^{N} X_i ,$$

где N – количество реализаций случайной величины X.

Оценка дисперсии случайной величины X определяется выражением

$$\tilde{D} = \frac{1}{N-1}\sum_{i=1}^{N} (X_i - \tilde{m})^2 .$$

В силу центральной предельной теоремы теории вероятностей при больших значениях количества реализаций N среднее арифметическое будет иметь распределение, близкое к нормальному закону [15] с математическим ожиданием

$$m[\tilde{m}] \approx \tilde{m}$$

и средним квадратическим отклонением

$$\sigma[\tilde{m}] \approx \frac{\sigma}{\sqrt{N}} ,$$

где σ – среднее квадратическое отклонение оцениваемого параметра.

При этом вероятность того, что оценка $\tilde{m}$ отклонится от своего математического ожидания меньше, чем на ε (доверительная вероятность), равна [15]

$$P(|\tilde{m} - m| < \varepsilon) \approx 2\Phi\left(\frac{\varepsilon}{\sigma[\tilde{m}]}\right), \qquad (4)$$

где $\Phi(x)$ – функция Лапласа, определяется выражением

$$\Phi(x) = \frac{1}{\sqrt{2\pi}}\int_0^x e^{-\frac{t^2}{2}} dt . \qquad (5)$$

Таким образом, при проведении экспериментальных исследований коллизионных свойств будем использовать методы статистической проверки гипотез и математической статистики.

1. Из генеральной совокупности случайной величины X сформируем выборку следующим образом:

– для среднестатистической оценки математического ожидания $m(n_1)$ и дисперсии $D(n_1)$ в качестве случайной величины выступает максимум $n_1(x_1, x_2)$ при которых выполняется равенство $h(x_1) = h(x_2)$, следовательно, выборку объема N: X_1, X_2, ..., X_N сформируем отбором N множеств, в каждом из которых содержится M пар элементов $x_1, x_2 \in A$, $x_1 \neq x_2$ и оценивается $n_1(x_1, x_2)$, т.е. общий объем формируемых пар элементов $x_1, x_2 \in A$, $x_1 \neq x_2$ составит NM;

– для среднестатистической оценки $m(n_2)$ и $D(n_2)$ в качестве случайной величины выступает максимум $n_2(x_1, y_1)$ при которых выполняется равенство $y_1 = h(x_1)$, следовательно, выборку объема N: X_1, X_2, ..., X_N сформируем отбором N множеств, в каждом из которых содержится M пар элементов $x_1 \in A$, $y_1 \in B$ и оценивается $n_2(x_1, y_1)$. Общий объем формируемых пар элементов $x_1 \in A$, $y_1 \in B$ составит NM;

– для среднестатистической оценки $m(n_3)$ и $D(n_3)$ в качестве случайной величины выступает максимум $n_3(x_1, x_2, y_1, y_2)$ при которых выполняются равенства $y_1 = h(x_1)$ и $y_2 = h(x_2)$, следовательно, выборку объема N: X_1, X_2, ..., X_N сформируем отбором N множеств, в каждом из которых содержится M четверок элементов $x_1, x_2 \in A$, $x_1 \neq x_2$, $y_1, y_2 \in B$ и оценивается $n_3(x_1, x_2, y_1, y_2)$, общий объем формируемых четверок составит NM.

2. При экспериментальных исследованиях коллизионных свойств хеширования будем оценивать среднее арифметическое $\tilde{m}(n_i)$ наблюдаемых значений максимумов n_i и дисперсию $\tilde{D}(n_i)$, $i = 1, 2, 3$.

3. Достоверность полученных среднестатистических оценок обоснуем следующим образом. Зафиксируем точность ε и рассчитаем значения функции Лапласа, которые, в соответствии с выражением (4), дадут соответствующие доверительные вероятности:

$$P(|\tilde{m}(n_i) - m(n_i)| < \varepsilon) \approx 2\Phi\left(\frac{\varepsilon}{\sigma[\tilde{m}(n_i)]}\right),$$

$$\sigma[\tilde{m}(n_i)] \approx \frac{\sqrt{\tilde{D}(n_i)}}{\sqrt{N}} .$$

При обратной постановке задачи, т.е. для фиксированной доверительной вероятности $P_д$ при объеме выборки N доверительный интервал определим следующим образом:

$$\tilde{m}(n_i) - t_р \cdot \sigma[\tilde{m}(n_i)] < m(n_i) < \tilde{m}(n_i) + t_р \cdot \sigma[\tilde{m}(n_i)] , \qquad (6)$$

где $t_р$ – корень уравнения $2\Phi(t_р) = P_д$.

Таким образом, предлагаемая методика, используя уменьшенные модели отдельных слоев преобразований, на основе оценки распределения столкновений формируемых образов позволяет экспериментально исследовать коллизионные свойства кодов аутентификации сообщений.

4. РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ И ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

С использованием разработанной уменьшенной модели UMAC (mini-UMAC) и методики статистического исследования коллизионных свойств кодов аутентификации сообщений проведем экспериментальную оценку распределения числа столкновений (коллизий) формируемых образов.

Поскольку в рассмотренной выше схеме UMAC на первом слое (при формировании

хеш-кода $Y_{\min i}$) используются семейства универсальных хеширующих функций, подробно исследуемые в работах [1-7], статистические исследования проведем только на втором слое (при формировании псевдослучайной подложки $Pad_{\min i}$) и на заключительном этапе формирования кодов аутентификации (после выполнения суммирования $Tag_{\min i} = Y_{\min i} \oplus Pad_{\min i}$). Именно на этих этапах, по нашему предположению и нарушаются свойства универсальности формируемых кодов аутентификации.

При проведении статистических исследований коллизионных свойств формируемых значений $Pad_{\min i}$ и $Tag_{\min i}$ для каждого эксперимента оценивались математические ожидания $m(n_1)$, $m(n_2)$ и $m(n_3)$, дисперсии $D(n_1)$, $D(n_2)$ и $D(n_3)$, а также для фиксированной точности $\varepsilon = 0{,}1$ рассчитывались соответствующие доверительные вероятности $P(|\tilde{m}(n_i) - m(n_i)| < \varepsilon)$. Исследования проводились над выборкой, объема $N = 100$, для формирования каждого элемента выборки рассчитывался максимум по множеству из $M = 1000$ кортежей элементов. Таким образом, общий объем формируемых наборов составил $NM = 10^5$.

Полученные результаты экспериментальных исследований сведены в табл. 2.

Таблица 2

Результаты экспериментальных исследований коллизионных свойств кодов аутентификации, сформированных с использованием mini-AES и mini-UMAC

	mini-AES, $Pad_{\min i}$	mini-UMAC, $Tag_{\min i}$
$\tilde{m}(n_1)$	–	4,23
$\tilde{D}(n_1)$	–	0,18
$P_д = P(\|\tilde{m}(n_1) - m(n_1)\| < \varepsilon)$	–	0,98
$\tilde{m}(n_2)$	6,68	4,78
$\tilde{D}(n_2)$	0,42	0,42
$P_д = P(\|\tilde{m}(n_2) - m(n_2)\| < \varepsilon)$	0,88	0,88
$\tilde{m}(n_3)$	0,19	5,31
$\tilde{D}(n_3)$	0,15	0,24
$P_д = P(\|\tilde{m}(n_3) - m(n_3)\| < \varepsilon)$	0,99	0,96

При исследовании коллизионных свойств кодов аутентификации, сформированных с использованием мини-версии шифра AES, число ключей, для которых выполняется равенство $h(x_1) = h(x_2)$, при всех испытаниях равнялось нулю, т.е. $n_1(x_1, x_2) = 0$ во всех $N = 100$ опытах. Этот результат объясняется следующим свойством. Шифр AES (как и его мини-версия), реализует биективное отображение множества открытых текстов во множество шифрограмм, т.е. для фиксированного ключа формируемые шифртексты, соответствующие различным открытым текстам, будут различны. Проводимые экспериментальные исследования по первому введенному критерию как раз и состояли в подсчете числа ключей, при которых наблюдается столкновение (коллизия) двух шифр-текстов, соответствующих двум различным открытым текстам, что невозможно по определению биективного шифра. В связи с этим статистические данные по первому критерию для мини-версии шифра AES в таблице 2 не приведены как не информативные.

Анализ приведенных в таблице 2 данных позволяет утверждать об адекватности полученных результатов и соответствии их статистическим свойствам всей генеральной совокупности данных. Для фиксированной точности $\varepsilon = 0{,}1$ получены высокие значения доверительной вероятности, что свидетельствует об обоснованности и достоверности полученных экспериментальных результатов.

Проанализируем полученные результаты статистических исследований коллизионных свойств кодов аутентификации сообщений, сравним полученные результаты среднестатистических оценок математических ожиданий $m(n_1)$, $m(n_2)$ и $m(n_3)$ числа правил хеширования, при которых выполняются равенства (1), (2) и (3), соответственно, с теоретическим оценками: числом $P_{кол} \cdot |H|$ (для первого критерия), с числом $|H|/|B|$ (для второго критерия) и числом $P_{кол} \cdot H$ (для третьего критерия).

Рассмотрим *первый критерий*, по которому оценивается число правил хеширования, при которых существует коллизия (совпадение кодов аутентификации) для двух произвольных входных последовательностей. В соответствии с теоретическими оценками эта величина ограничена сверху числом $P_{кол} \cdot |H|$. Конкретизируем эту (теоретическую) оценку для кодов аутентификации, сформированных с использованием mini-AES и mini-UMAC.

Мощность ключевого множества для mini-AES и mini-UMAC составляет $|H| = 2^{16}$, мощность множества формируемых кодов аутентификации также составляет $|B| = 2^{16}$. Если использовать верхнюю оценку вероятности коллизий как обратную величину мощности формируемых кодов аутентификации $P_{кол} = 2^{-16}$ получим $n_1(x_1, x_2) \le P_{кол} \cdot |H| = 1$. Для мини-версии шифра AES это условие выполняется (обосновывается биективностью шифрующего преобразования), однако коллизионные свойства mini-UMAC существенно уступают этой верхней теоретической оценке. Фактически, число коллизий выше теоретической границы более чем в четыре раза и это положение подтверждено с высокой доверительной вероятностью $P_д = P(|\tilde{m}(n_1) - m(n_1)| < 0{,}1) > 0{,}98$.

Рассмотрим *второй критерий*, по которому оценивается число правил хеширования, при которых для произвольной входной последовательности значение кода аутентификации не изменяется. В соответствии с теоретическими оценками эта величина для кодов аутентификации, сформированных с использованием mini-AES и mini-UMAC, ограничена сверху числом $|H|/|B|=1$. Полученные экспериментальные результаты свидетельствуют, что коллизионные свойства кодов аутентификации, сформированных с использованием mini-AES и mini-UMAC, не удовлетворяют второму критерию, число ключей, при которых для произвольной входной последовательности значение кода аутентификации не изменяется в несколько раз превышает теоретическую оценку для универсального хеширования.

В соответствии с *третьим критерием* оценивается число правил хеширования, при которых для двух произвольных входных последовательностей соответствующие им значения кода аутентификации не изменяются. Теоретическая оценка этой величины для универсального хеширования ограничена сверху числом $P_{\text{кол}}|H|$, что при использовании верхней оценки вероятности коллизий $P_{\text{кол}}=2^{-16}$ дает $n_3(x_1,x_2,y_1,y_2)\le P_{\text{кол}}\cdot|H|=1$. Значения, приведенные в таблице 2, свидетельствуют о том, что коллизионные свойства кодов аутентификации, сформированных с использованием mini-AES, удовлетворяют третьему критерию. В тоже время число ключей mini-UMAC, при которых для двух произвольных входных последовательностей соответствующие им значения кода аутентификации не изменяются, более чем в пять раз выше верхней теоретической оценки.

ВЫВОДЫ

Таким образом, из полученных результатов статистических исследований коллизионных свойств кодов аутентификации сообщений, сформированных с использованием mini-AES и mini-UMAC, можно сделать следующие важные в прикладном отношении выводы:

– криптографический слой формирования кодов аутентификации сообщений (mini-AES) удовлетворяет свойствам универсального хеширования, вероятность коллизии формируемых хеш-образов не превосходит наперед заданной величины (первый критерий).

Это объясняется, прежде всего, тем, что шифрование неповторяющегося (уникального) для всех информационных сообщений значения *Nonce* приводит к формированию множества неповторяющихся (уникальных) для всех информационных сообщений псевдослучайных подложек *Pad*.

Другими словами, формирование псевдослучайных подложек *Pad* осуществляется в результате биективного отображения множества неповторяющихся (уникальных) для всех информационных сообщений значений *Nonce*, в результате чего коллизии (столкновения) подложек *Pad* отсутствуют по определению. В тоже время, данный слой преобразований не удовлетворяет свойствам строго универсального хеширования (не выполняется второй критерий) (см. табл. 2). Кроме того, обеспечение свойств универсального хеширования на этом слое предполагает формирование и передачу неповторяющегося для каждого сообщения значения *Nonce*, что требует дополнительных временных и программно-аппаратных затрат;

– результат формирования кодов аутентификации сообщений по схеме mini-UMAC не удовлетворяет свойствам как универсального хеширования, так и, тем более, свойствами строго универсального хеширования. Это объясняется тем, что схема с простым суммированием по модулю два (XOR) двух результатов универсального хеширования не всегда сохраняет свойства универсального хеширования.

Поясним последний вывод на примере. Пусть первый и второй слой схемы формирования кодов аутентификации обладают свойствами универсального хеширования. Условно обозначим процесс такого хеширования в виде таблиц 3 и 4, где столбцами обозначены информационные сообщения $M_1, M_2, \ldots, M_n$, а строками – правила хеширования (h_i и g_j, соответственно), заданные (параметризированные) соответствующими секретными ключами. В ячейках таблиц содержатся результаты хеширования, т.е. искомые хеш-коды.

Таблица 3

	M_1	M_2	M_3	...	M_n
h_1	$h_1(M_1)$	$h_1(M_2)$	$h_1(M_3)$	...	$h_1(M_n)$
h_2	$h_2(M_1)$	$h_2(M_2)$	$h_2(M_3)$	...	$h_2(M_n)$
h_3	$h_3(M_1)$	$h_3(M_2)$	$h_3(M_3)$	...	$h_3(M_n)$
...	...	...	...	...	...
h_k	$h_k(M_1)$	$h_k(M_2)$	$h_k(M_3)$	...	$h_k(M_n)$

Таблица 4

	$M1$	M_2	M_3	...	M_n
g_1	$g_1(M_1)$	$g_1(M_2)$	$g_1(M_3)$	...	$g_1(M_n)$
g_2	$g_2(M_1)$	$g_2(M_2)$	$g_2(M_3)$	...	$g_2(M_n)$
g_3	$g_3(M_1)$	$g_3(M_2)$	$g_3(M_3)$	...	$g_3(M_n)$
	...	...	...	...	...
g_k	$g_k(M_1)$	$g_k(M_2)$	$g_k(M_3)$	...	$g_k(M_n)$

Если каждая пара правил хеширования h_i и g_i задается одним секретным ключом K_i, тогда результирующая схема с простым суммированием по модулю два (XOR) двух результатов универсального хеширования информационных сообщений может быть представлена табл. 5.

Таким образом, общее число правил хеширования не изменилось (по сравнению с числом правил хеширования для функций $h(x)$ и $g(x)$, соответственно), оно определяется мощностью множества используемых секретных ключевых данных. Каждый секретный ключ K_i задает

Таблица 5

		M_1	M_2	M_3	...	M_n
K_1	h_1, g_1	$h_1(M_1)\oplus g_1(M_1)$	$h_1(M_2)\oplus g_1(M_2)$	$h_1(M_3)\oplus g_1(M_3)$	...	$h_1(M_n)\oplus g_1(M_n)$
K_2	h_2, g_2	$h_2(M_1)\oplus g_2(M_1)$	$h_2(M_2)\oplus g_2(M_2)$	$h_2(M_3)\oplus g_2(M_3)$	...	$h_2(M_n)\oplus g_2(M_n)$
K_3	h_3, g_3	$h_3(M_1)\oplus g_3(M_1)$	$h_3(M_2)\oplus g_3(M_2)$	$h_3(M_3)\oplus g_3(M_3)$	...	$h_3(M_n)\oplus g_3(M_n)$
...	...	...	...	...	...	...
K_k	h_k, g_k	$h_k(M_1)\oplus g_k(M_1)$	$h_k(M_2)\oplus g_k(M_2)$	$h_k(M_3)\oplus g_k(M_3)$	...	$h_k(M_n)\oplus g_k(M_n)$

(параметризирует) два правила h_i и g_i, которые применяются к каждому информационному сообщению, подлежащему хешированию. Результат преобразования представлен в соответствующих ячейках таблицы 5 как результат суммирования по модулю 2 (XOR) значений $h_i(M_j)$ и $g_i(M_j)$.

Очевидно, что коллизия хеш-кодов будет наблюдаться для всех сообщений M_i и M_j, для которых выполняется равенство:

$$h_w(M_i)\oplus g_w(M_i) = h_w(M_j)\oplus g_w(M_j). \qquad (7)$$

Даже если функции h_w и g_w для сообщений M_i и M_j не вызывают коллизию, т.е., если

$$h_w(M_i) \neq h_w(M_j)$$

и

$$g_w(M_i) \neq g_w(M_j)$$

равенство (7) все равно может выполняться, и число правил (и число соответствующих ключей), вызывающих коллизию в результирующей схеме, возрастет. Это событие будет достоверным (произойдет наверняка), например, в случае, если

$$h_w(M_i) = h_w(M_j)$$

и

$$g_w(M_i) = g_w(M_j)$$

Таким образом, схема с простым суммированием по модулю два (XOR) двух результатов универсального хеширования в общем случае не обеспечивает сохранение свойств универсального хеширования. Коллизионные свойства кодов аутентификации сообщений снижаются и, как показывает анализ таблицы 2, не удовлетворяют поставленным требованиям.

Таким образом, нарушение коллизионных свойств универсального хеширования в схеме mini-UMAC (после применения криптографического слоя преобразования) следует считать экспериментально доказанным.

Перспективным направлением дальнейших исследований является разработка методов построения криптографически стойких схем формирования кодов аутентификации с обеспечением высоких коллизионных свойств универсального хеширования. Одним из перспективных направлений в этом смысле является использование модулярных преобразований.

Литература

[1] *Black J.* "UMAC: Fast and provably secure message authentication", Advances in Cryptology / J. Black, S. Halevi H., Krawczyk, T. Krovetz, P. Rogaway. – CRYPTO '99, LNCS vol. 1666, PP. 216-233, Springer-Verlag, 1999.

[2] *T. Krovetz, P. Rogaway.* Fast universal hashing with small keys and no preprocessing", work in progress, 2000. – URL: http://www.cs.ucdavis.edu/~rogaway/umac

[3] *T. Krovetz, J. Black, S. Halevi, A. Hevia, H. Krawczyk, P. Rogaway.* UMAC -Message authentication code using universal hashing. IETF Internet Draft, draft-krovetz-umac-00.txt. – URL: www.cs.ucdavis.edu/~rogaway/umac, 2000.

[4] *Krovetz T.* UMAC -Message authentication code using universal hashing. IETF Internet Draft, draft-krovetz-umac-02.txt. – URL: www.cs.ucdavis.edu/~rogaway/umac, 2004.

[5] Final report of European project number IST-1999-12324, named New European Schemes for Signatures, Integrity, and Encryption, April 19, 2004 – Version 0.15 (beta), Springer-Verlag.

[6] *Krovetz T.* UMAC - Message authentication code using universal hashing, 2006. – URL: http://www.cs.ucdavis.edu/~rogaway/umac

[7] *Krovetz T.* Software-Optimized Universal Hashing and Message Authentication. Dissertation submitted in partial satisfaction of the requirements for the degree of doctor of philosophy. University Of California Davis. September 2000. – 269 p.

[8] *Carter J. L.* Universal classes of hash functions / J.L. Carter, M.N. Wegman // Computer and System Scince – 1979 – №18 – P. 143–154.

[9] *Wegman M. N.* New hash functions and their use in authentication and set equality / M. N.Wegman, J. L. Carter / Computer and System Scince – 1981 – № 22 – P. 265-279.

[10] A Description of Baby Rijndael // ISU CprE/Math 533; NTU ST765-U. – 2003.

[11] Raphael Chung-Wei Phan, "Mini Advanced Encryption Standard (Mini-AES): A testbed for Cryptanalysis Students", Cryptologia, XXVI (4), October 2002. – PP. 283-306.

[12] *Долгов В.И.* Исследование дифференциальных свойств мини-шифров Baby-ADE и Baby-AES / В.И. Долгов, А.А. Кузнецов, Р.В. Сергиенко, О.И. Олешко // Прикладная радиоэлектроника. – Х.: ХНУРЭ, 2009. – Т. 8, № 3. – С. 252–257.

[13] *Долгов В.И.* Подход к криптоанализу современных шифров / В.И. Долгов, И.В. Лисицкая, Р.В. Олейников. // Материалы второй международной конференции «Современные информационные системы. Проблемы и тенденции развития», Харьков-Туапсе, Украина, 2–5 октября. – 2007. – С. 435–436.

[14] *Сорока Л.С.* Исследование дифференциальных свойств блочно-симметричных шифров. / Сорока Л.С., Кузнецов А.А., Московченко И.В., Исаев С.А. // Системи обробки інформації. – Харків: ХУ ПС. – 2010 – Вип. 6(87). – С. 286–294.

[15] *Вентцель Е.С.* Теория вероятностей / Е.С. Вентцель. – М.: Государственное издательство физико-математической литературы, 1958 – 564 с.

Поступила в редколлегию 9.04.2012

Кузнецов Александр Александрович, доктор технических наук, профессор, профессор кафедры БИТ ХНУРЭ. *Область научных интересов:* теория помехоустойчивого кодирования, криптография и аутентификация.

Король Ольга Григорьевна, преподаватель кафедры информационных систем ХНЕУ. *Область научных интересов:* теория аутентификации, методы и вычислительные алгоритмы хеширования информации.

Евсеев Сергей Петрович, кандидат технических наук, с.н.с., доцент кафедры информационных систем ХНЕУ. *Область научных интересов:* теория кодирования, криптография и аутентификация.

УДК 681.3.06

Дослідження колізійних властивостей кодів автентифікації повідомлень UMAC / О.О. Кузнецов, О.Г. Король, С.П. Євсєев // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 171–183.

Розглядається алгоритм формування кодів автентифікації повідомлень UMAC, в основі якого лежить використання універсальних гешуючіх функцій. Пропонується зменшена модель UMAC (mini-umac) і методика статистичного дослідження колізійних властивостей формованих кодів автентифікації повідомлень. З використанням зменшеної моделі UMAC досліджуються колізійні властивості кодів автентифікації, показано, що застосування криптографічного перетворення (з використанням алгоритму AES) на завершальному етапі UMAC приводить до порушення властивостей універсального гешування.

Ключові слова: міні-UMAC, автентифікація, універсальна функція, коди автентичності, алгоритм AES.

Таб. 5. Бібліогр.: 15 найм.

UDC 681.3.06

Studying collision characteristics of authentication codes of messages UMAC / A.A. Kuznetsov, O.G. Korol, S.P. Evseev // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 171–183.

The paper considers the algorithm of forming authentication codes of messages UMAC which is based on use of universal hashing functions. A reduced model UMAC (mini-UMAC) and methods of statistical research of collision characteristics of formed authentication message codes are suggested. The collision characteristics of authentication codes are researched with the help of using the reduced model UMAC. It is shown that using cryptographic transformation (with the application of the AES algorithm) at the final UMAC stage results in violation of universal hashing properties.

Keywords: mini-UMAC, authentication, universal function, authentication codes, AES algorithm.

Tab. 5. Ref.: 15 items.

УДК 621.3.06

АНАЛІЗ ГЕНЕРАТОРА ПСЕВДОВИПАДКОВИХ ПОСЛІДОВНОСТЕЙ ЗАСНОВАНОГО НА БАГАТОРАЗОВОМУ ГЕШУВАННІ

Ю.І. ГОРБЕНКО, Д.Е. ХРЯПІН

Наводиться опис генератора псевдовипадкових послідовностей заснованого на багаторазовому гешуванні, визначеному в стандарті X9.98. Приводяться результати дослідження колізійних властивостей методу генерування. Продемонстровано доказ стійкості генератору проти атак різного виду та надаються рекомендації, щодо зміни конструкції генератора. Наводяться результати визначення швидкості генерації псевдовипадкових послідовностей. Надаються висновки, щодо використання методу генерування в системах захисту інформації.

Ключові слова: детермінований генератор випадкових послідовностей, гешування, статистичний портрет, швидкість генерування послідовностей.

ВСТУП

Суттєвими складовими інформаційних та інформаційно-телекомунікаційних систем, від властивостей яких залежить якість надання криптографічних послуг, є засоби генерування ключів та параметрів. Існуючі методи генерування ключів та параметрів можна розділити на два великих класи – генератори випадкових послідовностей (ГВП) та генератори псевдовипадкових послідовностей(ГПВП). Для двійкового алфавіту ГПВП в [1, 2] названо детермінованим генератором випадкових бітів (ДГВБ). Обидва вказані класи генераторів знаходять застосування, але більш широко використовуються ДГВП, по крайній мірі в частині інтенсивності їх використання.

До ДГВП висувається ряд вимог. В загальному ДГВП та ПВП, що ним генеруються, повинні задовольняти вимогам необоротності, нерозрізнюваності та непередбачуваності [1, 2]. Згідно вказаних вимог період повторення lz повинен бути не менше припустимого ln, ентропія джерела ключів $H(k)$ та безпечний час $t_б$ також не менше припустимих значень, тобто $H(k) \geq Hp(K)$ та $t_б \geq t_р$. Крім того, реалізація ПВП Yi повинна задовольняти вимогам випадковості, рівноймовірності, незалежності та однозначності, а також забезпечувати генерування бітів з допустимою складністю (швидкодією).

Проведений аналіз показав, що ДГВП, які засновані на функціях гешування, мають ряд переваг. Так, ДГВП можуть використовувати будь-яку криптографічну функцію гешування за умови забезпечення достатньої ентропії для початкового значення. Але для таких ДГВП необхідно генерувати, в тому числі згідно ключів (ключа), символи прообразів з довільним алфавітом послідовності прообразу, з завідомо заданим періодом повторення $l_р$, допустимою швидкодією (складністю) v генерування символів та стійкістю проти визначення закону генерування ДГВП, яку називають непередбачуваністю.

1. ІНСТАЛЯЦІЯ ПОЧАТКОВОГО СТАНУ ТА ГЕНЕРУВАННЯ ПВП

В [1, 2] розглянуті методи генерування ПВП, які спираються на двокаскадну схему. В них перший каскад забезпечує генерування послідовності з необхідним алфавітом та періодом повторення, а другий забезпечує необхідні властивості непередбачуваності, нерозрізнюваності та необоротності. В стандарті X9.98 [1] наведене криптографічне забезпечення системи NTRU, в тому числі математична модель двох каскадного ДГВП, в якому обидва каскади грунтуються на використанні функцій гешування. В цілому в стандарті визначено чотири рівня безпеки, кожен з яких визначається мінімальною ентропією початкової або повторної ініціалізації. Ці рівні безпеки можуть бути реалізовані засобом використання функцій гешування. В таблиці 1 наведено класифікацію та визначення функцій гешування у залежності від рівня безпеки. Рекомендовано використовувати функції гешування сімейства SHA (Secure hash) [2, 3].

Таблиця 1

Рівні безпеки у залежності від довжини геш значення

Рівень захисту	112	128	192	256
SHA-1	+	+	–	–
SHA-224	+	+	+	-
SHA-256	+	+	+	+
SHA-384	+	+	+	+
SHA-512	+	+	+	+

Вхідними даними алгоритму для генерування ПВП є:

– бітова строчка, з використанням якої отримують внутрішній стан генератора;

– додаткові дані – бітова строчка, яка призначена для внесення додаткової ентропії в внутрішній стан на протязі циклу роботи генератора;

– необхідний рівень захисту – ціле число яке вказує на рівень захисту, який повинен забезпечувати ДГВП (фактично вибір функції гешування та максимальної кількості запитів на генерацію за один сеанс);

– необхідна кількість бітів для генерації – ціле число, яке вказує на довжину бітової строчки, що повинна бути генерована на виході ДГВП.

З урахуванням вказаного алгоритм генерації може бути представленим у наступному вигляді:

1) *Ініціалізація внутрішнього стану*. Під час ініціалізації перевіряються усі значення

параметрів довжин на відповідність заданому рівню захисту, розраховуються наступні значення внутрішнього стану:

$$V = hash(0x01 \| seed\_length \| entropy\_input \| \| personalization\_string) \| \| \| hash(0x02 \| seed\_length \| \| entropy\_input \| personalization\_string) \| .... \| \| hash(seed\_length / hash\_outlen) \| \| \| seed\_length \| \| entropy\_input \| personalization\_string)$$

$$C = hash(0x01 \| seed\_length \| 0x00 \| V) \| \| hash(0x02 \| seed\_length \| 0x00 \| V) \| \| ..... \| hash(seed\_length / hash\_outlen \| \| seed\_length \| 0x00 \| V)$$

2) *Генерування вихідного значення.* Після отримання запиту на генерація та генерування ПВП на вихід подається наступне значення.

$$output = hash(V) \| hash(V+1) \| ... \| hash(V + + requested\_number\_of\_bits / hash\_outlen)$$

3) *Оновлення внутрішнього стану.* Якщо кількість оброблених запитів більш ніж максимально дозволена встановленим рівнем захисту, то внутрішній стан оновлюється за наступними згідно таких перетворень

$$V = hash(0x01 \| seed\_length \| 0x01 \| V \| entropy\_input \| \| additional\_input) \| \| \| hash(0x02 \| seed\_length \| 0x01 \| \| V \| entropy\_input \| additional\_input) \| ..... \| \| hash(seed\_length / hash\_outlen \| seed\_length \| 0x01 \| \| V \| entropy\_input \| additional\_input)$$

$$C = hash(0x01 \| seed\_length \| 0x00 \| V) \| \| hash(0x02 \| seed\_length \| 0x00 \| V) \| \| .... \| hash(seed\_length / hash\_outlen \| \| seed\_length \| 0x00 \| V)$$

В цілому перетворення, що подані 1) – 3) можна представити в вигляді алгоритму, блок-схема якого наведеної на рис. 1.

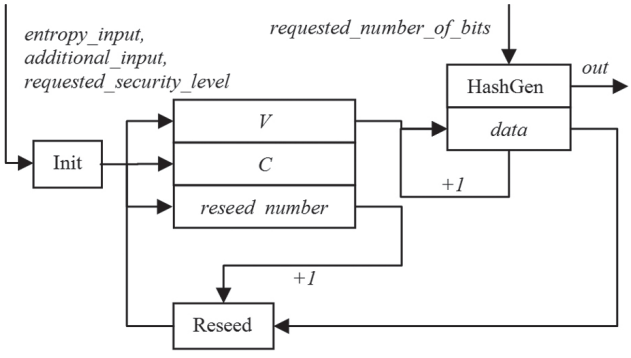


Рис. 1. Блок-схема генератора

1. АНАЛІЗ СТІЙКОСТІ ГЕНЕРАТОРА ПРОТИ АТАК

Проведемо аналіз стійкості ГПВП, що представлений на рис. 1, поклавши що він застосовується для генерації ключових даних, загально системних параметрів та даних в механізмах встановлення, узгодження, транспортування тощо ключів. Для вказаних криптографічних додатків було розроблено багато методів криптографічного аналізу генераторів. Тому актуальною є задача оцінки криптографічних якостей нерозрізнюваності, необоротності непередбачуваності та періоду повторення ПВП.

Для доказу стійкості генератора, що заснований на багаторазовому гешуванні, докажемо теореми для загального випадку (генератор заснованій на перетвореннях в підгрупах полів Галуа), а потім виконаємо змаштабування отриманих результатів на ДНВП, що досліджується. При розгляді будемо використовувати модель випадкового оракулу. В рамках цієї моделі криптоаналітик може знаходитися в декількох початкових станах, які будемо ранжувати за критерієм кількості відомої інформації про систему, тобто її ентропію. Криптоаналітик може находитись в наступних початкових умовах:

– відомі тільки загальносистемні параметри;
– відомі ЗСП та одне вихідне значення;
– відомі ЗСП та n вихідні значення;
– відомі ЗСП, n вихідних значений та один прообраз вихідного значення;
– відомі ЗСП, n вихідних значень и m прообразів вихідних значень.

Криптоаналітик може діяти, ставлячи перед собою одну з наступних цілей:

– отримати певне значення ;
– отримати значення сеансового ключа ;
– отримати значення начального ключа

Теорема 1 [1]. Якщо криптоаналітик володіючи максимальними знаннями про ДГВП на основі перетворень в підгрупі поля Галуа намагається отримати певне значення, то оракул, який він використовує для досягнення цією мети, можна використовувати для криптоаналізу циклової функції.

Доказ. Спираючись на структурну схему ДГВП на основі перетворень в підгрупі поля Галуа, що наведена на рис. 2, вихідне значення можна отримати використовуючи такий вираз

$$a_i = g^{K_k} f(a_{i-1}) \bmod p . \quad (1)$$

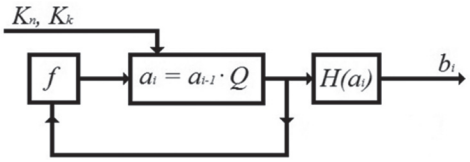


Рис. 2. Спрощена структурна схема ДГВП

Далі, вираз (1) можна перетворити до наступного вигляду:

$$g^{K_k} = \frac{a_i}{f(a_{i-1})}, \quad (2)$$

$$a_{i+k} = \frac{a_i f(a_{i+k-1})}{f(a_{i-1})} . \quad (3)$$

Аналіз (2) показує, що за умови використання не захищеної циклічної функції, складність отримання значення пошуку зводиться до

пошуку двох прообразів геш-значень. Такі задачі носять поліноміальний характер. При цьому, використовуючи (3), можна обчислити спираючись на знання трьох прообразів, довільний попередній, або наступний стан генератора. Так, якщо у аналітика є оракул, то можна отримати

$$f(a_{i-1}) = g^{K_k} \cdot a_i .$$

Але якщо в якості циклової функції використовується криптографічно стійка функція (стійке криптографічне перетворення, наприклад симетричне шифрування), то оракул аналітика виконує ефективний криптоаналіз симетричного шифру. Що у випадку використання, наприклад симетричного шифру, суперечить сучасному стану вирішення цієї задачі.

Теорема 2 [1]. Якщо криптоаналітик володіючи максимальними знаннями про ДГВП на основі перетворень в підгрупі поля Галуа намагається отримати значення, то оракул, який він використовує для досягнення цією мети, можна використовувати для вирішення дискретного логарифму в полі.

Доказ. Оракул, який може визначити сеансовий ключ генератора заснованого на перетвореннях в підгрупі поля Галуа та гешуванні можна використати для вирішення дискретного логарифму. Для цього на вхід оракулу необхідно подати ряд значень, що обчислюються за правилом:

$$a_0 = g^{K_k} ,$$

$$a_i = g^{K_k} \cdot f(a_{i-1}) .$$

Ця послідовність емітує послідовності, яку було генеровано ДГВП, у якого початковий і ключ сеансу не відрізняються. Таким чином оракул криптоаналітика, вирішує задачу дискретного логарифму в полі, що суперечить сучасному стану вирішення цієї задачі.

Таким чином загальна двокаскадна конструкція генератора є стійкою проти усіх можливих видів атак. Але використання менш стійкого перетворення на першому каскаді, зменшує рівень стійкості до поліноміального.

3. АНАЛІЗ ШВИДКОДІЇ ТА СТАТИСТИЧНИХ ВЛАСТИВОСТЕЙ ДГВП

Однією з основних вимог, що практично висуваються до ДГВП, та зрозуміло і до ПВП, є складність (швидкодія) її генерування. Для дослідження швидкодії було розроблено програмну модель ДГВП згідно математичної моделі, що мається в стандарті X9.98. В табл. 2 наведені дані відносно швидкодії такого ДГВП.

Порівняння даних табл. 2 з даними відносно швидкодії інших ДГВП такого ж призначення [4, 5], дозволяє зробити висновок, що вказаний генератор задовольняє вимогам відносно його до швидкодії. В той же час існує можливість оптимізації ДГВП по критерію швидкодії, в тому числі на рівні програмної моделі.

Таблиця 2

Швидкість генерації (Мгбіт /сек)

Геш-функція	SHA-1	SHA-256	SHA-512
Швикдоція	15,5	21,4	47,5

Зрозуміло, що основним критерієм безпеки відносно ДГВП є його характеристики нерозрізнюваності. Для дослідження якості нерозрізнюваності було використано NIST STS 800-22 [5]. На рис. 3–5 наведені фазові портрети ПВП

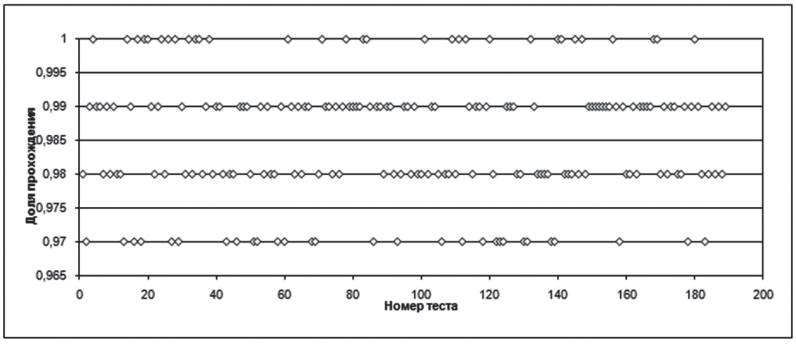


Рис. 3. ДГВП з геш-функцією SHA-1

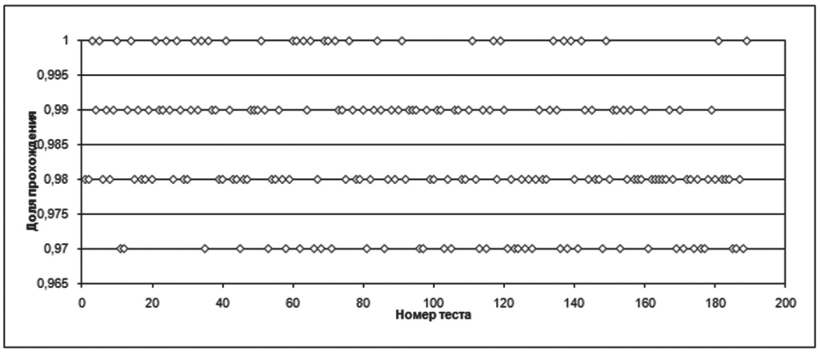


Рис. 4. ДГВП з геш-функцією SHA-256

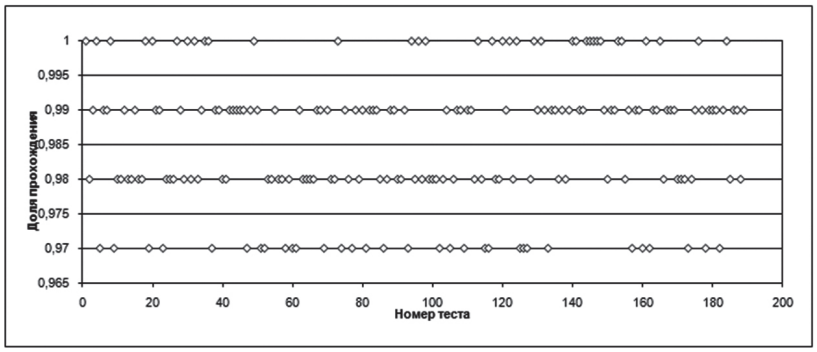


Рис. 5. ДГВП з геш-функцією SHA-512

ДГВП з різними функціями гешування, з застосуванням методики, що визначена в NIST STS. Їх порівняльний аналіз з даними відносно інших ДГВП [4], дозволяє зробити висновок що пропонуємий в стандарті X9.98 ДГВП є одним із кращих по критерію нерозрізнюваності.

ВИСНОВКИ

1) Більшість існуючих генераторів мають недоліки пов'язані з швидкодією генерації, або рівнем криптографічної стійкості методу генерації послідовностей. Перспективними для подальшого розвитку та актуальними для досліджень є генератори що засновуються на двокаскадній схемі. А саме отримання послідовності з необхідними властивостями та подальше гешування елементів цієї послідовності.

2) Представлений ДГВП має двокаскадну схему, яка складається з каскаду підготовки елементів та каскаду вхідного перетворення.

3) Проаналізований метод володіє високими показниками швидкодії, які зумовлені тим, що в якості фінального перетворення використовується геш функція.

4) Представлений ДГВП має високу криптографічну стійкість від атак різного вигляду. Це зумовлено тим, що криптографічна його стійкість спирається на складність вирішення криптографічних задачі пошуку прообразу геш значення.

5) Порівнюваний генератор володіє колізійними властивостями які задовольняють найжорсткішим вимогам дійсних стандартів в галузі безпеки інформації України та світовим вимогам.

Література

[1] American national standard for financial services (ANSI) X9.98. Lattice Based Polynomial Public Key Establishment Algorithm for the Financial Services Industry. – approved 2010-10-15. – ANSI, 2010. – 297 pages.

[2] Federal Information Processing Standards Publication (FIPS PUB) 180-2. Secure hash standard. – approved 2002-08-01. – NIST, 2002. – 76 pages.

[3] Federal Information Processing Standards Publication (FIPS PUB) 180-3. Secure hash standard (SHS). – approved 2008-10. – NIST, 2008. – 32 pages.

[4] *Горбенко І.Д.* Метод побудування випадкових бітів на основі спарювання точок кліптичних кривих / І.Д. Горбенко, Н.В. Шапочка, К.А. Погребняк // Журн. Прикладная радиоэлектроника. – 2010. – Т. 9, №3. – С. 386-394.

[5] NIST SP 800-22. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. – 2001-09. – 164 pages.

Надійшла до редколегії 12.04.2012

Хряпін Дмитро Едуардович, магістрант кафедри БІТ ХНУРЕ. Область наукових інтересів: аналіз асиметричних криптосистем, методів генерування випадкових послідовностей і функцій гешування, генератори ПВП, асиметричні криптопримітиви в групі точок еліптичних кривих.

Горбенко Юрій Іванович, кандидат технічних наук, технічний директор ЗАТ «ІІТ», науковий співробітник НІЦ «Z» каф. БІТ ХНУРЕ. Область наукових інтересів: захист інформації в інформаційно-телекомунікаційних системах.

УДК 621.3.06

Анализ генератора псевдослучайных последовательностей основанного на многократном хешировании / Ю.И. Горбенко, Д.Э. Хряпин // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 184–187.

Проводится анализ генератора псевдослучайных последовательностей основанного на многократном хешировании. Приводятся результаты оценки криптографической стойкости генератора против атак разного вида, показатели быстродействия генерации последовательностей, и их статистические свойства.

Ключевые слова: детерминированный генератор случайных последовательностей, хеширование, статистический портрет, скорость генерации последовательностей..

Табл. 2. Ил. 5. Библиогр.: 5 назв.

UDC 621.3.06

Analysis of pseudorandom sequence generator based on multiple hashing / Yu.I. Gorbenko, D.E. Khriapin // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 184–187.

Analysis of a pseudorandom number generator based on multiple hashing is performed. The paper presents results of evaluating cryptographic resistance of a generator against various kinds of attacks, indications of sequence generation speed and their statistical properties.

Keywords: determined random sequence generator, hashing, statistical portrait, sequence generation speed.

Tab. 2. Fig. 5. Ref.: 5 items.

УДК 681.324.067

ПОРІВНЯЛЬНИЙ АНАЛІЗ АЛГОРИТМІВ ГЕНЕРАЦІЇ ПСЕВДОВИПАДКОВИХ ПОСЛІДОВНОСТЕЙ

І.Д. ГОРБЕНКО, Р.І. МОРДВІНОВ

На сучасному етапі розвитку інформаційних технологій актуальними є проблеми захисту інформації. Захисту інформації, що зберігається в електронному вигляді, реалізується криптографічними методами. Для функціонування цих методів необхідно виконувати управління ключовими даними, а саме генерування ключів та параметрів.

Ключові слова: випадкова послідовність, генератор випадкових послідовностей, псевдовипадкова послідовність, детермінований генератор випадкової послідовності.

ВСТУП

На сучасному етапі розвитку інформаційних технологій актуальними є питання захисту інформації. Основними методами захисту такої інформації є криптографічні методи. Однією з умов для функціонування криптографічних систем є управління ключовими даними, а саме генерація ключових даних і параметрів. Для цього необхідно використовувати генератори випадкових / псевдовипадкових послідовностей.

1. ДЕТЕРМІНОВАНІ ГЕНЕРАТОРИ ПСЕВДОВИПАДКОВИХ БІТ НА ОСНОВІ ГЕШ-ФУНКЦІЙ

Ідея побудови генератора псевдовипадкових біт (ГПВБ) на основі геш-функцій лежить у використанні необоротних геш-функцій, за допомогою яких на основі початкового значення виробляються псевдовипадкові біти (ПВБ). Дані ГПВБ можуть використовувати будь-які криптографічні геш-функції, відповідні ISO / IEC 10118-3, і можуть використовуватися в додатках, які вимагають різні рівень захисту, але за умови використання відповідної геш-функції та отримання достатньої ентропії для початкового значення.

До ГПВБ на основі геш-функцій висуваються наступні вимоги:

1. Вихідні дані для геш-функцій повинні бути випадковими і різними для різних вхідних даних.

2. Початкове значення повинне мати необхідну ентропію.

ГПВБ на основі геш-функцій може проектуватися для забезпечення різних рівнів захисту в залежності від геш-функції, яка використовується. Стійкість геш-функції в даних ГПВБ дорівнює розміру вихідного блоку. При цьому необхідно зазначити, що якщо геш-функція використовується як елемент криптографічного послуги, де необхідно враховувати стійкість до колізій, то стійкість вихідних даних геш-функції оцінюється половиною розміру вихідного блоку завдяки «парадоксу дня народження».

Довжина початкового значення має бути максимально наближеною до розміру блоку гешированних даних та рівнем стійкості захисту.

ГПВБ на основі геш-функцій вимагає використання геш-функцій кілька разів, включаючи процес ініціалізації і переініціалізації. На всьому життєвому циклі генератора повинна використовуватися одна і та ж геш-функція, яка повинна відповідати бажаної стійкості захисту криптографічного програми.

Перед початком роботи ГПВБ необхідно встановити в початковий стан. Для отримання початкового значення використовуються вхідні дані ентропії.

Початкове значення використовується для отримання елементів початкового стану, що складається з:

1. Значення (V), яке оновлюється при кожному виклику ГПВБ;

2. Константи (C), яка залежить від початкового значення;

3. Лічильник, який вказує число запитів ПВБ з моменту отримання нового значення ентропії;

4. Стійкість захисту реалізації ГПВБ;

5. Довжини початкового значення;

6. Показника, який вказує на необхідність забезпечення прямої секретності ГПВБ;

7. (За бажанням) Перетворення вхідних даних ентропії з використанням односторонньої функції для подальшого порівняння з новими вхідними даними ентропії під час переініціалізації ГПВБ. Це значення необхідно для переініціалізації ГПВБ.

2. ДЕТЕРМІНОВАНІ ГЕНЕРАТОРИ ПСЕВДОВИПАДКОВИХ БІТ НА ОСНОВІ ФУНКЦІЙ БЛОЧНОГО ШИФРУВАННЯ

ГПВБ на основі блочного шифрування можуть використовувати будь-які алгоритми блокового шифрування, які описані в стандарті ISO / IEC 18033-3, а також можуть використовуватися в криптографічних додатках, в яких використовуються різні рівні захисту.

Для всіх операцій блочного шифрування повинні використовуватися одні й ті ж алгоритми блокового шифрування і довжина ключа.

Алгоритм блокового шифрування та розмір ключа повинні відповідати вимогам захисту програми.

Ініціалізація і переініціалізація ГПВБ на основі блочного шифрування повинна складатися з отримання початкового значення з необхідною кількістю ентропії. Вхідні дані ентропії використовуються для формування початкового значення, яке потім використовується для отримання елементів початкового стану ГПВБ. Початковий стан складається з:

1. Значення (V), яке оновлюється до кожним формуванням вихідного значення ПВБ;

2. Ключа, який оновлюється кожного разу, коли генерується задане число вихідних блоків;

3. Довжини ключа, які використовуються алгоритмом блокового шифрування;

4. Стійкість захисту ГПВБ;

5. Лічильник, який визначає число запитів, необхідних для генерації ПВБ з моменту ініціалізації або переініціалізціі;

6. Показника, що вказує на необхідність забезпечення ГПВБ стійкості до прогнозування.

В якості функції шифрування використовується функція Block_cipher (Key, data), де Key – ключ, data – дані, які шифруються. В якості алгоритмів шифрування використовуються такі алгоритми: AES, ГОСТ-28147-89, DES, TDES.

AES – Advanced encryption standart – міжнародний стандарт, прийнятий державним стандартом США. Може використовуватися з довжинами блоку та ключа 128, 192, 256 біт.

ГОСТ 28147-89 – державний стандарт Росії. Використовує Фейстел-подібну систему. Розмір блоку дорівнює 64 біта, розмір ключа – 256 біт.

DES – був державним стандартом США до прийняття AES. Використовує Фейстел-подібну систему шифрування. Розмір блоку 64 біта, розмір ключа – 64 біта (56 біт ключа і 8 біт перевірки).

TDES - модифікація шифру DES. Являє собою скомпоновані 3 шифру DES. Як шифрування може використовуватися шифрування, розшифрування і зашифрування з використанням різних ключів. Розшифрування відбувається в зворотному порядку. Розмір блоку і ключів відповідає стандартному DES.

3. ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

У табл. 1 наведено оцінку складності формування ПВБ з використанням даний ГПВБ в загальному вигляді через математичні операції. При цьому необхідно виділити 2 етапи щодо генерації ПВБ – підготовчий етап і безпосередньо етап генерації.

Для тестування даних ГПВБ використовувалася методика тестування NIST STS.

Для тестування було обрано такі параметри:

1. Довжина послідовності, яка тестується $n = 106$ біт;

2. Кількість послідовностей для тестування $m = 100$;

3. Рівень значущості $\alpha = 0{,}01$;

4. Кількість тестів $q = 189$.

Таким чином розмір вибірки для тестування дорівнює 108 біт, а статистичний портрет генератора має 18900 значень ймовірності P.

В ідеальному випадку при $m = 100$ і $\alpha = 0{,}01$ може бути відкинута тільки одна послідовність зі ста, тобто коефіцієнт проходження кожного тесту дорівнює 99%. Однак це занадто жорстке правило. Тому використовується правило на основі довіреної інтервалу. При цьому нижня межа дорівнює 0,96015. Результати тестування наведені в табл. 2.

Таблиця 1

Оцінка складності ГПВБ

	ГПВБ на основі геш-функцій	ГПВБ на основі функцій шифрування
Додавання	1 + Cycles	0
Додавання за модулем 2	0	2
Додавання за модулем	2	2 + Cycles
Конкатенація	5 + Cycles	2 + Cycles
Геш-функція	3 + Cycles	0
Функція шифрування	0	2 + Cycles

* cycles – відношення кількості біт, заданих для генерації до довжини вихідного блоку шифру або геш-функції.

Таблиця 2

Результати тестування послідовностей

Генератор	Кількість тестів, які успішно пройдені при рівні $\alpha = 0{,}99$	Кількість тестів, які успішно пройдені при рівні $\alpha = 0{,}96015$	Швидкість (Мбіт/с)
BBS	134 (71%)	189 (100%)	
SHA1	132 (69%)	188 (99%)	21
SHA2 256	130 (68%)	187 (98%)	15,3
SHA2 384	133 (70%)	189 (100%)	15,8
SHA2 512	141 (74%)	189 (100%)	16,7
AES	138 (73%)	189 (100%)	21,4
DES	132 (69%)	188 (99%)	19,5
ГОСТ 28-147	132 (69%)	188 (99%)	16,2
TDES	135 (71%)	189 (100%)	16,1

В результаті аналізу отриманих даних були зроблені наступні висновки: генератори на основі SHA2-512, AES і TDES мають кращі результати по NIST STS, які вище результатів BBS. Серед цих генераторів кращу швидкість показав генератор на AES (дослідження проводилися на Intel Celeron 2.8GHz). Таким чином кращим серед цих генераторів є генератор на блочному шифрі AES.

Література

[1] ISO/IEC 18031:2005 Information technology — Security techniques — Random bit generation.

Надійшла до редколегії 16.04.2012

Горбенко Іван Дмитрович – д.т.н., професор, завідувач кафедри безпеки інформаційних технологій Харківського національного університету радіоелектроніки, головний конструктор АТ «Інститут інформаційних технологій». Область наукових інтересів: криптографічні системи та протоколи, проектування та розробка систем, комплексів та засобів криптографічного захисту інформації.

Мордвінов Руслан Ігоревич – аспірант кафедри БІТ Харківського національного університету радіоелектроніки. Область наукових інтересів: розробка та застосування методів генерації випадкових послідовностей.

УДК 681.324.067

Сравнительный анализ алгоритмов генерации псевдослучайных последовательностей / И.Д. Горбенко, Р.И. Мордвинов / Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 188–190.

На современном этапе развития информационных технологий актуальными являются проблемы защиты информации. Защита информации, которая хранится в электронном виде, реализуются криптографическими методами. Для функционирования таких методов необходимо использовать управление ключевыми данными, а именно генерация ключей и параметров.

Ключевые слова: случайная последовательность, генератор случайных последовательностей, псевдослучайная последовательность, генератор псевдослучайной последовательности.

Табл. 2. Библиогр.: 1 назв.

UDC 681.324.067

Comparative analysis of pseudorandom sequence generation algorithms / I.D. Gorbenko, R.I. Mordvinov // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 188–190.

At the present-day stage of IT progress information security becomes increasingly important. The main methods of information security are cryptographic methods. To function such methods require using key data management, namely, generation of keys and parameters.

Keywords: random sequence, random sequence generator, pseudorandom sequence, pseudorandom sequence generator.

Tab. 2. Ref.: 1 items

УДК 004.056

МЕТОДЫ ГЕНЕРАЦИИ ПСЕВДОСЛУЧАНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ И ОЦЕНКА ИХ СВОЙСТВ

А.А.ЗАМУЛА, Д.А. СЕМЧЕНКО

Рассмотрены некоторые методы генерации псевдослучайных последовательностей (ПСП), приведены оценки их свойств на случайность с использованием Diehard тестов. На основе математических моделей методов генерации ПСП и результатов исследований статистических свойств последовательностей посредством программной реализации, описаны преимущества и недостатки каждого из рассматриваемых методов. Сделаны выводы о необходимости использования перспективных методов генерации ПСП, обеспечивающих реализацию требований к последовательностям с точки зрения криптографических свойств формируемых последовательностей символов.

Ключевые слова: генератор, псевдослучайная последовательность, метод, конгруэнтный, Xorshift, Фибоначчи, Diehard, сид.

ВВЕДЕНИЕ

На сегодняшний день разработано большое количество различных видов генераторов псевдослучайных последовательностей (ПСП). Одной из основных проблем при генерации ПСП, в частности, для криптографических приложений, является поддержание определенных криптографических свойств. При генерации последовательности чисел $X = x_1, x_2, ...$ необходимо убедиться, что случайная величина X обладает равномерным законом распределения, её реализации случайны и независимы [1]. Для проверки гипотезы о законе распределения используются статистические критерии χ^2 Пирсона, Колмогорова-Смирнова, Мизеса ω^2 [2] и др.

Математическая модель некоторых генераторов ПСП, использует: множество целых чисел Z, представленных 32-х разрядными машинными словами; сид (под сидом понимается начальное значение генератора) z; а также функциию f, определенную на множестве Z [3]. При случайном выборе сида z из Z, последовательность псевдослучайных чисел может быть получена путем многократного применения функции f:

$$f(z), f^2(z), f^3(z)...,$$

где $f^2(z)$ означает $f(f(z))$, а $f^3(z)$ означает $f(f^2(z))$ и так далее.

Однако, для генераторов ПСП, которые отвечают более строгим требованиям (большой период, m_i-ичное основание алфавита, структурная скрытность), множество Z представляется множеством всех m-кортежей $(x_1, x_2, ..., x_m)$ 32-х разрядных целых чисел, а функция f преобразует один из таких m-кортежей в другой.

Если f является биективной функцией (функция $f: X \to Y$ является биективной тогда и только тогда, когда существует обратная функция $f^{-1}: Y \to X$ такая, что $\forall x \in X$, $f^{-1}(f(x)) = x$ и $\forall y \in Y$, $f^{-1}(f(y)) = y$) определенной над множеством Z, то для любого сида z выбранного равномерно из Z, случайная переменная $f(z)$, полученная многократным применением функции f: $f(z), f^2(z), f^3(z)...$ также будет равномерно распределена по Z.

1. КОНГРУЭНТНЫЙ МЕТОД ГЕНЕРАЦИИ ПСП

Наиболее распространенным методом генерации ПСП является конгруэнтный метод, который для генерации ПСП использует правило:

$$x_n = (ax_{n-1} + k) \bmod m,$$

где m – модуль, a – множитель, k – аддитивная константа и x_0 - инициализирующий случайный сид.

Если a является примитивным корнем простого числа p, а x_0 является случайным сидом из множества $Z = \{1, 2, 3, ..., p-1\}$, то последовательность, порожденная $x_n = (ax_{n-1} + k) \bmod m$, будет строго периодическая, с периодом $p-1$, и каждый элемент этой последовательности будет равномерной случайной величиной на множестве Z, однако при этом элементы такой последовательности не будут независимыми между собой.

Если взять четыре последовательных числа x, y, z и w, сгенерированных линейным конгруэнтным генератором ПСП с множителем a, то [4-5]:

– точка (x, y, z) попадает на решетку точек, сгенерированных линейных комбинаций точек $(1, a, a^2)$, $(0, m, 0)$, $(0, 0, m)$ с целочисленными коэффициентами;

– аналогично, любая точка (x, y) будет попадать на решетку сгенерированную всеми линейными комбинациями точек $(1, a)$ и $(0, m)$ с целочисленными коэффициентами;

– точка (x, y, z, w) в четырехмерном пространстве будет попадать на решетку целочисленных комбинаций четырех точек $(1, a, a^2, a^3)$, $(0, m, 0, 0)$, $(0, 0, m, 0)$, $(0, 0, 0m)$.

Предположим, что α, β, τ – какие-либо три точки на плоскости с координатами, которые являются последовательными результатами генератора ПСП. Тогда определитель матрицы 2×2 с рядами $(\beta - \alpha)$ и $(\tau - a)$ будет давать объем

параллелепипеда, определенного этими тремя точками и объем параллелепипеда должен быть кратен m. Таким образом, НОД пяти или шести таких определителей будет равен m и, в этом случае, можно определить аддитивную константу k и множитель a.

Гистограмма распределения элементов последовательности, сгенерированной конгруэнтным генератором ПСП, представлена на рис. 1.

Рис. 1. Гистограмма распределения элементов последовательности конгруэнтного генератора ПСП

На рис. 1 по оси X - EBCDIC (*Extended Binary Coded Decimal Interchange Code*) коды символов 00h,01h,02h,...0Ah,0Fh,10h, 11h,..., FEh,FFh, а по оси Y – частота появления символа в последовательности, выраженная в процентах.

Проведенное моделирование конгруэнтного генератора ПСП, показало, что такой генератор не отвечает требованиям определенным тестом DIEHARD [6], в частности, тестам: BINARY RANK TEST для 6x8 матриц, BITSTREAM TEST (OVERLAPPING 20-кортежей), OPSO, OQSO и DNA, COUNT-THE-1's TEST.

2. XORSHIFT МЕТОД ГЕНЕРАЦИИ ПСП

Рассматривая 32-х (или 64-х) битное целое число как элемент векторного пространства с компонентами в поле $\bmod 2$, сложение двух векторов может быть реализовано как исключающее или (XOR) операция, что в сочетании с операцией сдвига, может быть использовано для создания некоторых линейных преобразований над этим векторным пространством. Множество сидов Z является множеством всех ненулевых двоичных векторов 1×32, а f является линейной трансформацией на множестве Z, представленной двоичной матрицей T 32×32, где T – невырожденная.

Для случайного сида $y\in Z$ последовательность $yT, yT^2, yT^3...$ будет иметь период $2^{32}-1$ тогда и только тогда, когда порядок матрицы T равен $2^{32}-1$ в группе 32×32 невырожденных двоичных матриц. Если $T=(I+L^a)\cdot(I+R^b)\cdot(I+L^c)$, где L – матрица, которая дает сдвиг влево на 1 (в C, $y^{\wedge}=(y<<1)$, таким образом yL^a в C – есть $y^{\wedge}=(y<<a)$, то можно получить простой и быстрый способ для формирования матричного произведения [7]. Матрица R, являющаяся транспонированной L, дает сдвиг вправо на единицу. Таким образом, для $T=(I+L^a)\cdot(I+R^b)\cdot(I+L^c)$, для случайного 32-битного сида y из Z, каждый новый y в последовательности $yT, yT^2, yT^3...$ может быть получен посредством последовательного применения следующих трех инструкций: $y^{\wedge}=y<<13$, $y^{\wedge}=y>>17$, $y^{\wedge}=y<<5$.

Для 32-х (или 64-х) битных двоичных векторов отсутствуют двухсдвиговые матрицы $T=(I+L^a)\cdot(I+R^b)$, которые имеют полный период и нет односдвиговых, поэтому необходима 3-х сдвиговая матрица T. Существует точно 81 тройка [7] $[a,b,c]$, $a<c$, для которых 32×32 двоичная матрица $T=(I+L^a)\cdot(I+R^b)\cdot(I+L^c)$ имеет порядок: $2^{32}-1$.

Если матрица $T=(I+L^a)\cdot(I+R^b)\cdot(I+L^c)$ имеет полный период, то и $(I+L^c)\cdot(I+R^b)\cdot(I+L^a)$ и $(I+L^a)\cdot(I+L^c)\cdot(I+R^b)$ также имеют полный период, что дает возможность получить 4×81 матриц T с порядком $2^{32}-1$. Но тогда и транспонированная матрица каждой из них имеет полный период. Таким образом, могут быть получены 648 матриц.

Гистограмма распределения элементов последовательности, сгенерированной Xorshift генератором ПСП, представлена на рис. 2.

Рис. 2. Гистограмма распределения элементов последовательности Xorshift генератора ПСП

Проведенное моделирование Xorshift генератора ПСП, показало, что такой генератор не отвечает требованиям определенным тестом DIEHARD, в частности таким, как: BINARY RANK TEST для 6x8 матриц, OPSO, OQSO, COUNT-THE-1's TEST.

3. МЕТОД ГЕНЕРАЦИИ ПСП ФИБОНАЧЧИ С ЗАПАЗДЫВАНИЯМИ

Базовое рекуррентное соотношение для генераторов ПСП Фибоначчи [1] с запаздываниями представляется в виде $x_n=x_{n-r}\bullet x_{n-s}$ для r и s при $r>s$ [7]. Операция $\bullet$ определяется как бинарные отношения для пар элементов в некотором множестве χ, и множество сидов Z представляет собой множество r-кортежей $(x_1,x_2,...,x_r)$, где $x\in\chi$. Обычно χ является множеством 32-х битных целых чисел, а операция $\bullet$ является сложением или вычитанием по $\bmod 2^{32}$. Правило для $\bullet$ может быть основано на выражении элементов x, y из χ в форме $x=\pm 3^a, y=\pm 3^b \mod 2^{32}$ так, что $x\bullet y=\pm 3^{(a+b)\bmod 2^{30}}$ и верно рекуррентное правило для сложения $\bmod 2^{30}$. Для генераторов ПСП Фибоначчи с запаздываниями используется обозначение $F(r,s,\bullet)$. Для правильного выбора запаздываний r, s период $F(r,s,\pm \bmod 2^{32})$ должен быть 2^{32+r}, в то время как $F(r,s,\oplus)$ будет 2^r не зависимо от размера слова. Для правильного выбора $r>s$ период генератора $F(r,s,*$ нечетных по модулю $2^{32})$ будет 2^{30}.

В отличии от конгруэнтного и Xorshift методов генерации ПСП, метод генерации ПСП Фибоначчи с запаздываниями требует множество

сидов Z и функцию f, определенную над Z. Под Z в данном случае подразумевается множество кортежей $\{[x_1, x_2, ...x_r]\}$, где x из множества χ, на котором определено бинарное отношение и функция f :. $f([x_1, x_2, ..., x_r]) = [x_2, ..., x_r, x_1 \bullet x_{r-s+1}]$..

Реализация последовательностей Фибоначчи, с запаздываниями $r > s$ требует хранения таблицы последних r значений. Одно из самых полезных применений генератора ПСП Фибоначчи с запаздываниями является непосредственная генерация равномерных случайных чисел с плавающей запятой на интервале [0,1) .

Предположим, необходимо сгенерировать 64-х битные равномерные [0,1) случайные переменные, используя IEEE 754 стандарт (стандарт формата представления чисел с плавающей точкой, используемый как в программных реализациях арифметических действий, так и аппаратных реализациях): 1 знаковый бит, 11 битов экспоненты, 52 бита мантиссы с подразумеваемой ведущей единицей. Для указанного бинарного отношения $x \bullet y$ используют правило: если $x \geq y$, тогда $x - y$, в противном случае – $x - y + 1$. Если x и y – представление рациональных чисел $a / 2^{53}$ и $b / 2^{53}$ с плавающей запятой, то $x \bullet y$ будет давать в результате точное значение $c / 2^{53}$ с плавающей запятой, где $c = (x - y) \bmod 2^{53}$.

Гистограмма распределения элементов последовательности, сгенерированной генератором ПСП Фибоначчи с запаздываниями, представлена на рис. 3.

Рис. 3. Гистограмма распределения элементов последовательности сгенерированной генератором ПСП Фибоначчи с запаздываниями

Проведенное моделирование генератора ПСП Фибоначчи с запаздываниями, показало, что такой генератор не отвечает требованиям определенным тестом DIEHARD, в частности требованиям: OPSO, OQSO, DNA.

ВЫВОДЫ

Результаты проведенных исследований различных генераторов ПСП показали, что метод генерации Xorshift является более предпочтительным, чем конгруэнтный метод. Это обосновывается более простой технической реализацией и лучшей производительностью [8], а также свойствами случайности генерируемых последовательностей (показателями прохождения статических тестов DIEHARD). При использовании конгруэнтного метода достаточно легко определить аддитивную константу k и множитель a, поэтому этот метод генерации нельзя считать криптостойким, однако благодаря его простоте, он вполне может использоваться в открытых системах.

Анализ методов генерации ПСП показывает, что для обеспечения криптографических свойств ПСП, необходимо, чтобы сид был выбран независимо и базировался на случайных явлениях, либо процессах. Методы, обеспечивающие реализацию криптографических свойств ПСП, будут рассмотрены в следующих статьях.

Литература

[1] *Б. Шнаер*. Прикладная криптография / *Б. Шнаер* // 2-е издание.

[2] *Вентцель Е.С.*, Овчаров Л.А. “Теория вероятностей и ее инженерные приложения” 2-е изд. М.: Высшая школа, 2000.— 480 с.

[3] *Marsaglia, G* (2003). Seeds for random number generators.

[4] *Marsaglia, G* (1970). Regularities in congruential random number generators. Numerische Mathematic 16, 8-10.

[5] *Marsaglia, G* (1972). The structure of linear congruential sequences. In Applications of Number Theory to Numerical analysis, Z.K. Zaremba, ed. Academic Press, 249-285;

[6] The Marsaglia Random Number CDROM including the Diehard Battery of Tests of Randomness // [Электронныйресурс]: http://www.stat.fsu.edu/pub/diehard/.

[7] *Marsaglia, G*, 2003, Random number generators, Journal of Modern Applied Statistical Methods, No. 2.

[8] *Дональд Кнут*. Искусство программирования / Дональд Кнут // Получисленные алгоритмы. The Art of Computer Programming. – Vol.2. Seminumerical Algorithms. – 3-е изд. – М.: «Вильямс»/ – 2007. – С. 832.

Поступила в редколлегию 23.04.2012

Замула Александр Андреевич, профессор кафедры БИТ ХНУРЭ, кандидат технических наук, доцент. Область научных интересов: технологии защиты информации в информационно-телекоммуникационных системах.

Семченко Денис Александрович, аспирант кафедры БИТ ХНУРЭ. Область научных интересов: разработка и применение методов генерации псевдослучайных последовательностей, тестирование программного обеспечения.

УДК 004.056

Методи генерації псевдовипадкових послідовностей та оцінка їх властивостей / О. А. Замула, Д.О. Семченко // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 191–194.

Розглянуті деякі методи генерації псевдовипадкових послідовностей (ПСП), наведено оцінки їх властивостей на випадковість з використанням Diehard тестів. На основі математичних моделей методів генерації ПСП та результатів досліджень статистичних власти-

востей послідовностей за допомогою програмної реалізації, описані переваги та недоліки кожного з розглянутих методів. Зроблені висновки про необхідність використання перспективних методів генерації ПСП, що забезпечують реалізацію вимог до послідовностей з точки зору криптографічних властивостей формованих послідовностей символів.

Ключові слова: генератор, псевдовипадкова послідовність, метод, конгруентний, алгоритм Xorshift, коди Фібоначчі, Diehard тести, сід.

Іл. 3. Бібліогр.: 8 найм.

UDC 004.056

Methods of generating pseudorandom sequences and evaluation of their properties / A. A. Zamula, D.A. Semchenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 191–194.

The paper considers some methods for generating pseudo-random sequences (PRS), evaluates their properties with the Diehard tests. On the basis of mathematical models of PRS methods for generating and research results of the statistical properties of sequences through the program implementation the advantages and disadvantages of each of these methods are described. Conclusions about the need for promising methods of generating PRSs to ensure the satisfaction of the requirements for the sequences in terms of cryptographic properties of the generated sequences of symbols are made.

Keywords: generator, pseudorandom sequence, method, congruent, Xorshift algorithm, Fibonacci codes, Diehard test, seed.

Fig. 3. Ref.: 8 items.

АСИММЕТРИЧНЫЕ КРИПТОГРАФИЧЕСКИЕ ПРЕОБРАЗОВАНИЯ. СИНТЕЗ, АНАЛИЗ, СВОЙСТВА, ПРИМЕНЕНИЯ

УДК 004 056 55

ОБОСНОВАНИЕ И ИССЛЕДОВАНИЕ ПРАКТИЧЕСКОЙ РЕАЛИЗАЦИИ УЛУЧШЕННОГО АЛГОРИТМА ЦИФРОВОЙ ПОДПИСИ NTRUSIGN

Е.Г. КАЧКО, Д.С. БАЛАГУРА, Ю.И. ГОРБЕНКО

Излагаются результаты обоснования и исследования рациональной практической реализации асимметричного алгоритма цифровой подписи NTRU Sign по критерию минимизации прямых и обратных преобразований.

Ключевые слова: алгоритм NTRU цифровой подписи(Sign), выбор параметров, генерация асимметричной пары ключей, формирование и проверка, сложность алгоритмов формирования и проверки NTRU Sign.

ВВЕДЕНИЕ

В связи с широким применением в настоящее время тщательно исследованы многие аспекты теории и практики цифровых подписей (ЦП) в классе RSA, DSA, ECDSA преобразований, в том числе отечественные стандарты, например, ДСТУ ГОСТ 34 310-2009, ДСТУ 4145-2002[1]. Все эти алгоритмы имеют общий недостаток – они существенно проигрывают в сложности (скорости) преобразований симметричным крипто алгоритмам [1–2]. Кроме этого, при появлении новых математических методов и существенном росте производительности крипто аналитических систем, криптографическая стойкость этих алгоритмов вызывает сомнение. Для повышения стойкости разработчики систем постоянно увеличивают размеры общесистемных параметров для этих алгоритмов. Особенно эта проблема остро встает в связи с разработками, связанными с созданием нейрокомпьютеров, в которых параллельно будут выполняться не только потоки, но и отдельные фрагменты программы (т.н. мелкозернистый параллелизм). Еще большую опасность несет возможное создание квантовых систем крипто анализа, которые будут в состоянии осуществлять атаки типа полное раскрытие для крипто преобразований, сложность которых носит субэкспоненциальный характер. Поэтому важными являются задачи поиска крипто преобразований, которые позволили бы, с одной стороны существенно увеличить скорость криптографических операций, а с другой стороны сохранить криптографическую стойкость. Последние исследования позволяют сделать вывод, что указанным требования могут удовлетворять крыпто преобразования в кольцах срезанных полиномов (NTRU)[2].

Целью статьи является обоснование сущности и разработка рациональной реализации алгоритма NTRU для ЦП, а также сравнения его вычислительной сложности с известными аналогичными асимметричными алгоритмами. При этом, вопросы оптимальности не ставятся, подразумевается некоторое повышение скорости, и такая реализация называется рациональной(улучшенной). Вопросы криптографической стойкости не рассматриваются, а только в отношении сложности криптопреобразований идет ориентация на работы [2-5].

1. ЦИФРОВАЯ ПОДПИСЬ

1.1. Параметры и алгоритмы

Основными этапами реализации ЦП является выбор общих параметров, генерация ключевой пары, а также основные- выработка и проверка ЦП.

Параметры цифровой подписи. Согласно [3] используются следующие параметры ЦП:

N – размер срезанного кольца R, которое определяет максимальную степень полинома (N–1), значения этих параметров составляют 439 и 743 [4];

p, q – малый и больший модули преобразования соответственно, которые являются взаимно простыми числами (для ЦП и для шифрования равны 3, 2048). Кроме того, параметр q определяет интервал, которому должны принадлежать все коэффициенты многочленов, использующихся в криптосистеме. Так, пространство сообщений L_M определяется как:

$$L_M = \{M(x) \in R\},$$

при этом коэффициенты полиномов сообщений лежат в диапазоне

$$[-(q-1)/2,(q-1)/2];$$

дополнительные параметры, которые рассматриваются ниже.

Личный ключ. В отличие от шифрования [2], личный ключ для ЦП содержит 4 полинома:

$f(x), g(x)$ (или f, g) – полиномы по модулю $p = 3$, которые должны иметь обратные элементы. При выборе этих полиномов для обеспечения максимальной безопасности, рекомендуется выбирать только такие, коэффициенты которых имеют заданное число 1 и –1. Эти значения также включаются в список параметров. Обозначим d_f (определяет количество 1, –1 в полиноме f) и d_g (определяет количество 1, –1 в полиноме g).

F, G – полиномы, которые удовлетворяют уравнению:

$$f * G - g * F = q. \quad (1)$$

Необходимо заметить, что значений F, G, удовлетворяющих уравнению (1) множество. Из этого множества следует выбирать наименьшие. Еще одно требование к этим полиномам – нормы векторов для них должны удовлетворять заданному значению, которое обозначается *KeyNormBound* и является параметром. Как следует из [3], всегда можно выбрать эти значения так, чтобы их коэффициенты не превосходили $\sqrt{\frac{N}{12}}$. Для значения *N*, равного 439, максимальное значение коэффициента равно 7, а для *N*, равного 743, максимальное значение коэффициента равно 8. Для минимизации коэффициентов полиномов F, G может использоваться итерационный алгоритм, в котором количество итераций для выбора ограничивается значением параметра *MaxAdjustment*.

Для формирования ЦП достаточно использовать один из полиномов пары (f, g) и один полином из пары (F, G). Выбор элементов пары определяется типом личного ключа. Тип ключа обозначается *basisType* и принимает 2 значения: "*standard*" или "*transpose*". В дальнейшем выбирается тип ключа *transpose*, который позволяет ускорить формирование ключевых данных и ЦП без потери криптографической стойкости[3].

В отличие от алгоритмов ЦП, которые нашли широкое распространение (RSA, DSA, ECDSA, ДСТУ 4145-2004,...), в NTRU открытый ключ используется не только для проверки, но и для ее создания. Поэтому открытый ключ либо вычисляется в процессе выработки ЦП, либо входит в структуру с личным ключом. Последнее более эффективно с точки зрения вычислительной сложности операции создания ЦП. Необходимость использования открытого ключа при создании ЦП связана с тем, что не все ЦП, которые получаются в соответствии с алгоритмом создания, могут быть проверены. ЦП считается «хорошей» не только в случае, если она удовлетворяет определенным математическим соотношениям, но и если норма вектора, соответствующего полиному, полученному в качестве ЦП, удовлетворяет определенным соотношениям. Предельное значение нормы определяется параметром *NormBound*. Поэтому после выработки ЦП она проверяется, и, только в случае успешной проверки, возвращается в качестве значения ЦП. Значение нормы вычисляется по формуле[6]:

$$\|s\|^2 = \sum_{i=0}^{N-1} s_i^2 - \frac{1}{N}\left(\sum_{i=0}^{N-1} s_i\right)^2.$$

При этом, ограничивается количество попыток, которые используются для создания ЦП. Параметр, который определяет число попыток, обозначается *SignFailTolerance*.

Для увеличения криптографической стойкости, создание ЦП выполняется в несколько этапов. Для каждого этапа используется внутренний личный и открытый ключи. Для каждого очередного шага вычисляется новое значение подписываемых данных. Для формирования ЦП на всех этапах необходимо иметь соответствующее количество личных ключей. Поэтому вместо обычной функции генерации личного ключа используется цикл генерации. Для проверки ЦП на стороне получателя используется последний открытый ключ. Количество личных ключей, рекомендуемое для формирования ЦП, задается параметром *perturbationBases*. В работе принимается значение этого параметра равным 1, что соответствует формированию ЦП в 2 этапа:

(for ($i = 0$; $i <= perturbationBases$; $++i$)).

Заметим, что многоуровневое формирование ЦП приводит к увеличению значения нормы на величину $\sqrt{perturbationBases + 1}$.

Открытый ключ. Задается полиномом по модулю *q*. При задании этого полинома используются только положительные числа, отрицательные значения заменяются *q*-значением.

Таким образом, в качестве дополнительных параметров ЦП используются: d_f (задает количество 1, –1 в полиноме f) и d_g (задает количество 1, –1 в полиноме g); *KeyNormBound*, *NormBound* – значения норм векторов, которые соответствуют полиномам (F, G) и ЦП s; *MaxAdjustment* – число итераций для минимизации полиномов F, G; *basisType* – определяет, какие компоненты пар (f, g), (F, G) используются при создании ЦП; *SignFailTolerance* – максимальное число итераций для получения «хорошей» подписи; *perturbationBases* – количество этапов формирования ЦП.

Общие параметры. В табл. 1 заданы значения общих параметров для N = 439 и 743 [4]

Таблица 1

Параметры для полиномов для $N = 439$ и 743

Параметр	N = 439	743
d_f^1 (d_g)	146	248
keyNormBound	280	360
NormBound	400	
MaxAdjustment	439	743
basisType	TRANSPOSE	TRANSPOSE
SignFailTolerance	100	100
perturbationBases	1	1

[1] Значение $d_f(d_g)$ задает количество – 1, количество 1 на 1 больше.

При вычислении ключевой пары необходимо вычислить результанты [5] ключевых полиномов (f,g)) и полинома (X^N-1). Результант – это значение определителя, порядок которого равен 2^N-1 и формируется так (показано для полиномов f и (X^N-1)):

$$\begin{bmatrix} f_{N-1} & f_{N-2} & f_{N-3} & \dots & f_0 & 0 & 0 & 0 & \dots & 0 \\ 0 & f_{N-1} & f_{N-2} & \dots & f_1 & f_0 & 0 & 0 & \dots & 0 \\ \dots & & & & & & & & & \\ 0 & 0 & 0 & \dots & f_{N-1} & f_{N-2} & f_{N-3} & f_{N-4} & \dots & f_0 \\ 1 & 0 & 0 & \dots & 0 & -1 & 0 & 0 & \dots & 0 \\ \dots & & & & & & & & & \end{bmatrix}$$

1.2 Генерация ключей цифровой подписи

Генерация ключей для одного этапа[2]

1. Случайно выбрать полином f, содержащий d_f единиц (1), d_f минус единиц (–1) и остальные 0.

2. Вычислить обратный элемент для $f_{inv}=f^{-1}$, т.е. такой элемент, что $f_{inv}*f=1$ в $(Z/qZ)[X]/(X^N-1)$.

3. Если обратный элемент не существует, то перейти на шаг 1.

4. Вычислить результант $\mathrm{Re}s_f(f,X^N-1)$, а также полином p_f, удовлетворяющий уравнению:

$$p_f*f=\mathrm{Re}s_f(f,Z[X]/X^N-1).$$

5. Случайно выбрать полином g, содержащий d_g единиц (1), d_g минус единиц (–1) и остальные 0.

6. Вычислить обратный элемент для $g_{inv}=g^{-1}$, т.е. такой элемент, что $g_{inv}*g=1$ в $(Z/qZ)[X]/(X^N-1)$. Если обратный элемент не существует, то перейти на шаг 4.

7. Вычислить результант $\mathrm{Re}s_g$ для полинома g и полинома X^N-1, и полином p_g, удовлетворяющий уравнению:

$$p_g*g=\mathrm{Re}s_g(g,Z[X]/X^N-1).$$

8. Решить диафантово уравнение с помощью расширенной теоремы Эвклида:

$$a*\mathrm{Re}s_f+b*\mathrm{Re}s_g=\gcd(\mathrm{Re}s_f,\mathrm{Re}s_g).$$

9. Если $\gcd(\mathrm{Re}s_f,\mathrm{Re}s_g)\neq 1$, то перейти на шаг 1 (наибольший общий делитель, не равный 1, говорит о наличии одинаковых корней у полиномов f, g, что недопустимо).

10. Вычислить полиномы F,G:

$$F=-p_g*B*q \text{ в } (Z/qZ)[X]/(X^N-1)$$

$$G=-p_g*A*q \text{ в } (Z/qZ)[X]/(X^N-1)$$

11. Реверсировать[3] полиномы f,g. Обозначим результат реверсирования f_{rev}, g_{rev} соответственно.

12. Вычислить $t=f*f_{rev}+g*g_{rev}$.

13. Вычислить результант $\mathrm{Re}s_t$ для полинома t и полинома x^N-1, и полином p_t, удовлетворяющий равенству $p_t*t-\mathrm{Re}s_t(Z[X]/(X^N-1))$.

14. Вычислить коэффициенты полнома *c*:

$$c=p_t*(f_{rev}*F+g_{rev}*G) \text{ в } (Z/qZ)[X]/(X^N-1).$$

15. Почленно разделить все элементы полинома *c* на значение $\mathrm{Re}s_t$ с округлением в большую сторону.

16. Вычислить

$$F-=c*f \text{ в } Z[X]/(X^N-1).$$

$$G-=c*g \text{ в } Z[X]/(X^N-1).$$

17. Минимизировать F, G (этот шаг алгоритма необязательный. Требуются дополнительные исследования для проверки целесообразности выполнения этого шага).

17.1 $u=f, v=g$;

17.2 Вычислить

$$E=2N\sum_{i=0}^{N-1}(f^2{}_i+g^2{}_i)-(f(1)+g(1))^2;$$

17.3 for (i=0; i < *MaxAdjustment*;)

a) Вычислить

$$D=4N\sum_{i=0}^{N-1}(F_if_i+G_ig_i)-2(F(1)+G(1))*(f(1)+g(1));$$

b) if (D>E) {F-=u; G-=v; ++i}; else if (D<-E)

{F+=u, G+=v; ++i}

c) u*=X; v*=X в $Z[X]/(X^N-1)$

18. Если

basisType = TRANSPOSE то $f'=F$.

basisType = STANDARD то $f'=F$.

19. Вычислить открытый ключ[4] $h=f\cdot f'$.

Ключи цифровой подписи. После выполнения этого алгоритма *perturbationBases*+1 раз получаем:

Набор личных ключей:

$$f_{perturbationBases}, f'_{perturbationBases}, h_{perturbationBases}$$

...

$$f_0, f'_0, h_0$$

Для значения *perturbationBases*=1 (см. табл. 1) получаем 2 набора ключевых данных:

$$f_1 f'_1, h_1$$

$$f_0 f'_0, h_0$$

Открытый ключ: $h=h_0$

1.3 Формирование цифровой подписи

Алгоритм формирования ЦП состоит из следующих этапов.

1. Преобразование подписываемого сообщения в полином.

[2] Число таких этапов равно *perturbationBases* + 1

[3] Полином называется реверсивным, если коэффициент с индексом 0 не изменяется, а остальные коэффициенты записываются в противоположном порядке, т.е. f1 ↔ fN-1, f2 ↔ fN-2, ...

[4] Полином f' имеет коэффициенты {0,1,-1} для типа TRANSPOSE и целые коэффициенты для типа STANDARD, поэтому генерация открытого ключа и цифровой подписи, которые используют значение f1, выполняется быстрее для типа TRANSPOSE.

2. Вычисление компонентов ЦП.

3. Преобразование компонентов цифровой подписи в строку байтов

1.3.1 *Преобразование подписываемого сообщения в полином*. Преобразование обычно использует значение хеша сообщения, что обеспечивает существенно различные полиномы для близких сообщений.

При реализации для сравнимости результатов используется алгоритм преобразования [4]:

Алгоритм преобразования сообщения в полином.

1. Определить количество байтов для задания одного элемента полинома: $B = \lceil \log_2 q / 8 \rceil$

2. for ($i = 0$; $i < n$; $++i$)

2.1 Вычислить хеш сообщения + i

2.2 Взять в полученном хеше младшие B байт

2.3 Сформировать значение коэффициента полинома равным $\log_2 q \% 8$ бит старшего байта и оставшиеся биты остальных байтов

1.3.2 П*реобразование полинома в массив байтов*

Для каждого коэффициента полинома выполняется его преобразование в битовую строку и конкатенация с предыдущей строкой битов.

1.4 Формирование цифровой подписи

Алгоритм формирования включает в себя формирование цифровой подписи для каждого этапа и суммирование цифровых подписей. Чтобы при проверке цифровой подписи можно было использовать только открытый ключ последнего этапа, на очередном этапе корректируется полином, для которого вычисляется ЦП.

Алгоритм.

Вход: Сообщение, представленное как полином (i)

Параметры: Личный ключ

Выход: При успешном завершении: Успех, Цифровая подпись s – полином, r – число попыток.

В случае ошибки: Ошибка, (за заданное число попыток (*SignFailTolerance*) не удалось сформировать цифровую подпись со значением нормы, не превосходящей *NormBound*).

1) *for (r = 0; r < SignFailTolerance; ++r)*

1.1) $s = 0$

1.2) *for (iLoop = perturbationBases; iLoop>=1; --iLoop)*

1.2.1) $s_{iLoop} = \lceil f_{iLoop} * i / q \rceil - \lceil f'_{iLoop} * i / q \rceil$ [5]

1.2.2) $s += s_{iLoop}$

1.2.3) $i = s_{iLoop} * (h_{iLoop} - h_{iLoop=1})$

1.2.4) $s_{iLoop} = \lceil f_{iLoop} * i / q \rceil - \lceil f'_{iLoop} * i / q \rceil$

1.3) $s += s_{iLoop}$

1.4) *if (норма s < NormBound) break;*

2) *if (r < SignFailTolerance)*

2.1) *Преобразование полинома s в строку байтов sByte*

2.2) *sByte+=байтовое представление (r)*

[5] Из этого шага алгоритма следует, что создание цифровой подписи выполняется быстрее для типа TRANSPOSE, чем для типа STANDARD (см. Предыдущую сноску)

1.5 Проверка цифровой подписи

Алгоритм проверки включает в себя преобразование ЦП в ее компоненты r, s, «расшифрование» ЦП с помощью открытого ключа, и проверку нормы для исходной ЦП и результата

Алгоритм.

Вход: Сообщение, представленное как полином (i)

Параметры: Открытый ключ h; Цифровая подпись (байтовая строка)

Выход: Успех; Ошибка

1) *Распаковка ЦП (s, r)*

2) *Формирование полинома i для заданных значений (s, r)*

3) $t = i - h * s$

4) $\| s \| \le NormBound \;\&\&\; \| t \| \le NormBound$

2. ИССЛЕДОВАНИЕ ВЫЧИСЛИТЕЛЬНОЙ СЛОЖНОСТИ АЛГОРИТМОВ СОЗДАНИЯ И ПРОВЕРКИ ЦИФРОВОЙ ПОДПИСИ

В данном разделе определяется вычислительная сложность функций выработки и проверки ЦП. Полученные результаты сравниваются с аналогичными результатами для формирования и проверки ЦП для RSA и ДСТУ 4145 – 2004 алгоритмов[1,2].

2.1 Выбор параметров цифровой подписи для сравнения.

При выборе параметров авторы исходили из крипто стойкости, обеспечиваемой при применении соответствующих алгоритмов. В [7, 8] определена зависимость уровня безопасности (k) от параметров цифровой подписи для алгоритмов RSA, ECC с простым и двоичными полями и NTRU при p = 3, *perturbationBases*=1 (табл. 2).

Таблица 2

Зависимость параметров алгоритмов от требуемого уровня безопасности

k	80	112	128	192	256
NTRU, N	157	197	223	313	349
RSA, n	1024	2048	3072	7680	15360
ECC, простое поле p	192	224	256	384	521
ECC, двоичное поле m	163	233	283	409	571

Из табл. 2 следует, что полиномы с N>349 для ЦП обеспечивают уровень безопасности больше, чем 256 и по уровню безопасности превосходят все рассмотренные алгоритмы с максимальными значениями параметров, которые используются в настоящее время.

2.2 Сравнение размеров и времен для различных алгоритмов цифровой подписи

Далее приведены результаты сравнения алгоритмов создания и проверки ЦП по двум критериям: размер ЦП (число байт) и время выполнения этих операций (табл. 3). Первая строчка таблицы соответствует результатам, полученным для подписи (N = 439) в Java реализации ([4]), остальные данные получены с помощью

библиотеки авторов (VS 2008, C, Intel R, Pentium (R), Dual CPU E2160 & 1.80 GHz, 0.99 ГБ ОЗУ). Увеличение быстродействия достигнуто за счет максимального использования параллелизма современных процессоров (SSE операции, многоядерность).

Таблица 3

Результаты сравнения размеров и времен для различных алгоритмов ЦП

Алгоритм	Размер цифровой подписи (байт)	Время создания (с)	Время проверки (с)
NTRU ($N = 439$), java [4]	610	2.4	2.4
NTRU ($N = 439$)	610	0,0033	0,0010
RSA[6], $n = 15360$	1920	88,81	0.214
ECC (ДСТУ 4145-2004), двоичное поле $m = 571$	144	0,042	0,169

Таким образом, использование алгоритма NTRU для ЦП позволяет уменьшить время создания и проверки цифровой подписи по сравнению с ДСТУ 4145-2004 более, чем в 10 раз при создания и более, чем в 100 раз при проверке. При этом размер ЦП увеличивается почти в 5 раз. Алгоритм RSA в этом случае проигрывает и по размеру подписи, и по времени ее выработки и проверки.

Для операций выработки и проверки ЦП время существенно различно (разница более, чем в 3 раза). Это связано с особенностями вычислений для обеих операций (см. соответствующие алгоритмы) и необходимостью проверки полученной ЦП (вычисление ее нормы).

Использование для проверки ЦП норм полиномов для ЦП и результата «шифрования» вместо тождеств, как для других алгоритмов ЦП, приводит к тому, что незначительное изменение открытого ключа иногда не приводят к ошибке проверки ЦП. Изменение остальных параметров (сообщения, личного ключа) приводит к ошибке при проверке ЦП.

В тоже время необходимо отметить, что применение ЦП в кольце срезанных полиномов ограничивается проблемой доказательства стойкости. Эта проблема должна рассматриваться отдельно.

Литература

[1] *Горбенко Ю.І., Горбенко І.Д.* Інфраструктури відкритих ключів . Системи ЕЦП. Теорія та практика. Харків. – Форт, 2010. – 593 с.

[2] Polynomial Public Key Establishment Algorithm for the Financial Services Industry.

[3] http://grouper.ieee.org/groups/1363/lattPK/submissions/EESS1v2.pdf

[4] http://grepcode.com/file/repo1.maven.org/maven2/net.sf.ntru/ntru/1.0/net/sf/ntru/sign/SignatureParameters.java

[5] http://ww2.math.uu.se/~svante/papers/sjN5.pdf

[6] http://www.securityinnovation.com/uploads/Crypto/NTRUSign-preV2.pdf

[7] http://www.securityinnovation.com/uploads/Crypto/lll25.pdf

[8] http://books.google.com.ua/books?id=z8nmMkUFqdwC&pg=PA486&lpg=PA486&dq=Security+level+RSA+ECDSA&source=bl&ots=As6_ELXJqP&sig=u67IGY8X4pZtq5kXMLdM8hadn_4&hl=ru#v=onepage&q=Security%20level%20RSA%20ECDSA&f=false

[6] Так как максимальный модуль, для которого определена процедура генерации ключей составляет 4096 (FIPS 186 -3) для определения времени создания цифровой подписи определяется время выполнения операции модульного возведения в степень для заданного значения модуля. При проверке цифровой подписи «открытый ключ»≈216.

Поступила в редколлегию 27.02.2012

Качко Елена Григорьевна, кандидат технических наук, профессор кафедры ПО ЭВМ ХНУРЭ. Область научных интересов: программные средства криптографических систем.

Балагура Дмитрий Сергеевич, кандидат технических наук, доцент кафедры Безопасности информационных технологий ХНУРЭ. Область научных интересов: защита информации, криптографические протоколы выработки и согласования ключей.

Горбенко Юрий Иванович, фото и сведения об авторе см. на с. 187.

УДК 004 056 55

Обґрунтування та дослідження практичної реалізації покращеного алгоритму цифрового підпису NTRUSign / О.Г. Качко, Д.С. Балагура, Ю.І. Горбенко // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 195–199.

Викладаються результати обґрунтування та дослідження раціональної практичної реалізації асиметричного алгоритму цифрового підпису NTRUSign за критерієм мінімізації прямих та зворотних перетворень.

Ключові слова: алгоритм NTRU цифрового підпису (Sign), вибір параметрів, генерація асиметричної пари ключів, формування та перевірка, складність алгоритмів формування та перевірки NTRUSign.

Табл. 03. Бібліогр.: 08 найм.

UDC 004 056 55

Grounding and researching the practical implementation of the improved algorithm of digital signature NTRUSign / E.G. Kachko, D.S. Balagura, Yu.I. Gorbenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 195–199.

Results of grounding and researching the rational practical implementation of the asymmetric algorithm of the digital signature NTRU sign by the criterion of minimizing forward and backword transformations.

Keywords: algorithm of NTRU digital signature (Sign), choice of parameters, generation of an assymetric pair of keys, forming and checking, complexity of algorithms of forming and checking NTRU Sign.

Tab. 03. Ref.: 08 items.

УДК 004.056.55

АНАЛІЗ КРИПТОГРАФІЧНИХ АЛГОРИТМІВ НА ІДЕНТИФІКАТОРАХ, ЩО ВИКОРИСТОВУЮТЬ АЛГЕБРАЇЧНІ РЕШІТКИ

І.Д. ГОРБЕНКО, Л.В. МАКУТОНІНА

Наводяться огляд та результати порівняльного аналізу основних алгоритмів криптографічних перетворень на ідентифікаторах, що використовують алгебраїчні решітки. Обґрунтовуються вибір параметрів, з точки зору забезпечення необхідного рівня стійкості. Наводяться умови та доказ безпеки для задачі навчання в моделі випадкового оракула з помилками.

Ключові слова: криптографічні системи на ідентифікаторах, алгебраїчні решітки, алгоритм зашифрування, алгоритм розшифрування, ідентифікатор користувача, прообраз вибірки, функція з секретом.

ВСТУП

Впровадження та використання існуючої інфраструктури відкритих ключів на сертифікатах, виявило ряд недоліків і проблемних питань, до яких можна віднести складність побудови, складність використання для кінцевого користувача і вартість. Альтернативою таким системам є криптосистеми з відкритим ключем на ідентифікаторах. Існує два перспективних напрями побудови криптографічних систем на ідентифікаторах (Identity-Based Encryption – IBE) – з використанням білінійних спаровувань на еліптичних кривих (ЕК) та з використанням решіток [6, 7, 18]. Існує також і третій підхід, який полягає в використанні елементарного теоретично-числового методу, запропонованого Коксом, Боне, Жентрі і Гамбургом [1, 2].

Метою даної статті є порівняльний аналіз основних схем на ідентифікаторах, що використовують алгебраїчні решітки та обґрунтування вибору кращої з них.

До недавнього часу майже всі IBE-конструкції були засновані на білінійних спарюваннях точок еліптичних кривих, але з появою роботи Джентрі, Вейкунтанатана і Пейкерта [3] особливий інтерес спостерігається з боку побудови криптографічних схем на ідентифікаторах, що використовують задачі на решітках. У даній статті описано і проаналізовано нові підходи побудови IBE-конструкцій, що використовують алгебраїчні решітки, та даються відповідні оцінки.

1. СХЕМА ДЖЕНТРИ, ВЕЙКУНТАНАТАНА І ПЕЙКЕРТА

Джентрі, Вейкунтанатана і Пейкерта показали [3], що можливо побудувати набір функцій з секретом з праобразу вибірки, що засновані на складності вирішення задачі про решітки. Запропонована схема використовує допоміжний алгоритм SampleD, та складається з трьох основних кроків:

1. Генерація функцій з секретом. Для будь-якого q (поліноміально обмеженого параметром безпеки n) та будь-якого $m \geq 5n\log_2 q$ існує імовірнісний алгоритм з поліноміальним часом виконання, який за вхідні значення приймає: 1^n виходів матриці $\mathbf{A} \in Z_q^{n\times m}$; повно ранговий набір $\mathbf{S} \subset \Lambda^{\perp}(\mathbf{A},q)$, з розподіленням A статистично близьким до рівномірного над $Z_q^{n\times m}$; $\|\mathbf{S}\| \leq L = m^{2,5}$. Переважна ймовірність належить $\mathbf{A}$ з рангом n.

2. Визначається функція $f_{\mathbf{A}}(\mathbf{e}) = \mathbf{Ae} \bmod qZ_q^{n\times m}$, ранг $R_n = Z_q^n$ та домен $D_n = \{\mathbf{e} \in Z^m : \|\mathbf{e}\| \leq s\sqrt{m}\}$. Використовуючи алгоритм SampleD може бути обране $D_{Z^m,s}$ розподілення по D_n, зі стандартним базисом для Z^m.

3. Застосовується інверсний алгоритм функції з секретом SampleISIS, з вхідними значеннями $(\mathbf{A},\mathbf{S},s,\mathbf{u})$, який обирає зі $f_{\mathbf{A}}^{-1}(\mathbf{u})$. Спочатку використовуються лінійні алгебраїчні функції для обчислення $\mathbf{t} \in Z^m$, такого, що $\mathbf{At} = \mathbf{u} \bmod q$. Після цього використовується алгоритм SampleD зі функцією з секретом $\mathbf{S}$ та вибірка $\mathbf{v}$ з розподілу $D_{\Lambda^{\perp},s,-\mathbf{t}}$, вихідним значенням алгоритму є $\mathbf{e} = \mathbf{t} + \mathbf{v}$.

Допоміжний алгоритм SampleD для кожного базису решітки $\mathbf{B} \in Z^{n\times k}$, кожного ідеалу $s \geq \|\mathbf{B}\| \cdot w(\sqrt{\log n})$ та кожного $\mathbf{c} \in IR^n$, виробляє вихідну послідовність SampleD$(\mathbf{B},s,\mathbf{c})$, з розподілом, в межах статистично незначної відстані $D_{L(\mathbf{B}),s,\mathbf{c}}$. Алгоритм виконується в поліноміальному часі з розміром вхідної послідовності n.

В основі схеми Джентрі, Вейкунтанатана і Пейкерта [3] лежить ідея обчислення підпису з геш-значення прообразу в заданому діапазоні. В IBE-схемах ключ розшифрування для заданого ідентифікатора можна розглядати як підпис уповноваженого на генерацію для даного ідентифікатора. Так, шляхом обчислення геш-значення ідентифікатора в діапазоні, через генерацію одного із прообразів, можливо обчислити ключ розшифрування. Хоча схема є концептуально простою, існують труднощі, які полягають в зіставленні випадкового оракула з ідентифікатором у заданому діапазоні.

Нехай n визначає параметр безпеки, та нехай $q = q(n)$ – просте, $m = m(n)$ – позитивне ціле число (аналогічно $O(n\log n)$), розмірність решітки.

1. Встановлення параметрів (Set-Up)

За допомогою алгоритму генерації функції з секретом, генерується матриця $\mathbf{A} \in Z_q^{n\times m}$ та функція з секретом $\mathbf{S} \subseteq \Lambda^{\perp}(\mathbf{A},q)$: для будь-якого $\mathbf{u} \in Z_q^n$

застосовується функція з секретом **S**, обирається $\mathbf{e} \in Z_q^m$ з набору всіх прообразів **u** над f.

Відкритими параметрами є: матриця **A**. Майстер ключ: **S**.

2. Вироблення секретного ключа користувача (Key-Gen).

Секретний ключ користувача генерується для відповідного ідентифікатора, який потім зберігається. На подальші запити ключа користувачу надається раніше генерований секретний ключ.

Під час генерації секретного ключа для ідентифікатора $id \in \{0,1\}^*$ використовується функція гешування $H:\{0,1\}^* \to Z_q^n$, що побудована для моделі з випадковим оракулом. Для даного id, нехай $\mathbf{u} = H(id)$, використовуючи майстер ключ **S** обчислюється **e** – прообраз **u** над f.

Секретний ключем для даного id є: **e**.

Нехай χ – розподіл в Z_q і χ^m – m-кратний добуток розподілу в Z_q^m. Розподіл χ має параметр r, що має бути обраним таким, щоб забезпечити складність розкриття, що дорівнює складності вирішення задачі «навчання з помилками» (learning with errors – LWE) над решітками [22].

3. Алгоритм зашифрування (Encrypt)

Зашифрування біта повідомлення b для ідентифікатора id виконується наступним чином:

a) нехай $\mathbf{u} = H(id) \in Z_q^n$;

b) рівномірно обирається **s** з Z_q^n;

c) встановлюється $\mathbf{p} = \mathbf{A}^{\mathrm{T}}\mathbf{s} + \mathbf{x}$, де **x** обирається з Z_q^m відповідно до розподілу χ^m;

d) обчислюється $c = \mathbf{u}^{\mathrm{T}}\mathbf{s} + x + b\lfloor q/2 \rfloor$, де x обирається з Z_q відповідно до χ.

Зашифрованим текстом є пара $(c, \mathbf{p})$.

4. Алгоритм розшифрування (Decrypt)

Розшифрування криптограми біта повідомлення $(c, \mathbf{p})$ для ідентифікатора id, за наявності отриманого від центру секретного ключа **e**, виконується наступним чином:

a) обчислюється $b' = c - \mathbf{e}^{\mathrm{T}}\mathbf{p} \in Z_q$;

b) якщо $b' \notin [0, \lfloor q/2 \rfloor) \bmod q$, вихідним значенням є 1; інакше – 0.

В алгоритмі зашифрування обирається вектор $\mathbf{p} = \mathbf{A}^{\mathrm{T}}\mathbf{s} + \mathbf{x}$ для рівномірно випадкового **s** і **x** обраного з χ^m. Даний вектор по суті є LWE-вектором. Вектор **u** є прикладом ще одного LWE-вектора для генерації $p = \mathbf{u}^{\mathrm{T}}\mathbf{s} + x$, де x обирається з χ. Значення p використовується як маска повідомлення b (перетворення $b \cdot \lfloor q/2 \rfloor$). Доки **u** є рівномірним для криптоаналітика криптограма також є рівномірною (в припущенні, що LWE є важко розв'язуваною). З цієї причини, ця схема має властивість анонімності зашифрованого тексту відносно до особи, яка його зашифрувала.

Більш формально, можна показати, що для правильно зашифрованого повідомлення, розшифрування може бути успішним з переважною ймовірністю, і, що схема CPA-безпечна і анонімна за умови, якщо LWE-задача з відповідно заданими параметрами є важко розв'язуваною.

Зашифрування здійснюється по одному біту за раз. В результаті, розмір зашифрованого тексту є досить великим. Певною мірою ця проблема може бути вирішена наступним чином. Припустимо, що повідомлення є k-бітним рядком. Ідентифікатори відображаються як H до k елементів $\mathbf{u}_1,\ldots,\mathbf{u}_k \in Z_q^n$, де $\mathbf{u}_i$ використовується для зашифрування i-го біта повідомлення. Випадкове **s** використовується для всіх бітів, і тому вектор $\mathbf{p} = \mathbf{A}^{\mathrm{T}}\mathbf{s} + \mathbf{x}$ залишається однаковим для кожного зашифрування. Відмітимо, що k-елементів з Z_q^m будуть частиною секретного ключа.

2. СХЕМА АГРАВАЛ, БОНЕ І БОЄНА

Схема запропонована Агравал, Боне і Боєном є різновидом схем ієрархічного шифрування на ідентифікаторах (HIBE). У попередній роботі Агравал, Боне і Боєна [4], ідентифікатор відображається в елемент **u** і, для обчислення прообразу, використовується функція з секретом **S**. Матриця **A** попередньо обчислюється і не залежить від ідентифікатора. На вищому рівні ця схема є подібною до класичної IBE-схеми Боне-Франкліна [5], де ідентифікатори відображаються в точки на ЕК і ключем розшифрування є результат s-разового скалярного множення цих точок.

Альтернативний підходом до побудови IBE-шифрування на решітках є заміна ролі діапазону точки на відображення матриці. У звичайній HIBE-схемі діапазон точки **u** є фіксованим і не залежить від ідентифікатора. Матриця **A** змінюється в $[\mathbf{A}_0 \,|\, \mathbf{A}_1 + H(id)\mathbf{B}]$, де $\mathbf{A}_0, \mathbf{A}_1, \mathbf{B}$ відносяться до відкритих параметрів, а $H: Z_q^n \to Z_q^{n \times n}$ є відкритою функцією, що відображає ідентифікатор (що вважається елементом Z_q^n) в матрицю. Функція H відображає Z_q^n до $Z_q^{n \times n}$, і задовольняє властивості «повно рангової різниці»: для будь-яких двох різних $\mathbf{u}, \mathbf{v} \in Z_q^n$, матриця $H(\mathbf{u}) - H(\mathbf{v})$ є повно ранговою.

Розглянемо матрицю $\mathbf{F} = [\mathbf{A}, \mathbf{AR} + \mathbf{B}]$, де $\mathbf{A}, \mathbf{B} \in Z_q^{n \times m}$ та $\mathbf{R} \in \{-1,1\}^{m \times m}$. Використовуються два метода відбору прообразів SampleLeft та SampleRight [6]. Метод SampleLeft працює зі скороченим базисом для **A**, і базується на узагальненому прообразі вибірки. Метод SampleRight базується на техніці делегації решітки. В реальних IBE-схемах використовується тільки метод SampleLeft, а метод SampleRight використовується тільки при моделюванні параметра безпеки.

Схема Агравал, Боне і Боєна [7] представлена наступним чином:

1. Встановлення параметрів (Set-Up)

Для даних n,q використовується метод генерації функції з секретом [8] для генерації матриці $\mathbf{A}_0 \in Z_q^{n \times m}$ та скорочений базис $\mathbf{S}_0$ для $\Lambda^{\perp}(\mathbf{A}_0)$. Обираються дві рівномірні випадкові матриці $\mathbf{A}_1, \mathbf{B} \in Z_q^{n \times m}$. Обирається рівномірно та випадково $\mathbf{u} \in Z_q^n$. Ідентифікатори вважаються елементами Z_q^n, для будь-якого ідентифікатора id, нехай:

$$\mathbf{F}_{id} = [\mathbf{A}_0, \mathbf{A}_1 + H(id)\mathbf{B}].$$

Відкритими параметрами є: $\mathbf{A}_0, \mathbf{A}_1, \mathbf{B}, \mathbf{u}$. Майстер ключ: $\mathbf{S}_0$.

2. Вироблення секретного ключа користувача (Key-Gen).

Нехай *id* – ідентифікатор, використаємо скорочений базис $\mathbf{S}_0 \subseteq \Lambda^{\perp}(\mathbf{A}_0)$ для обчислення прообразу $\mathbf{e}_{id} \in Z_q^{2m}$ над $\mathbf{u}$ для $\mathbf{F}_{id}$. Остання дія є алгоритмом SampleLeft, який узагальнює метод відбору прообразу. Тоді $\mathbf{F}_{id}\mathbf{e}_{id} = \mathbf{u}$.

Секретний ключем для даного *id* є: прообраз $\mathbf{e}_{id}$.

3. Алгоритм зашифрування (Encrypt).

Зашифрування біта повідомлення *b* для ідентифікатора *id* виконується наступним чином:

a) рівномірно випадково обирається $s \in Z_q^n$;

b) рівномірно випадково обирається матриця $\mathbf{R} \in \{-1,1\}^{m \times m}$;

c) обирається елемент шуму $x \in Z_q$ відповідно до розподілу $\bar{\Psi}_\alpha$;

d) обирається елемент шуму $y \in Z_q^m$ відповідно до розподілу $\bar{\Psi}_\alpha^m$;

e) встановлюється $\mathbf{z} = \mathbf{R}^{\mathrm{T}}\mathbf{y}$;

f) обчислюється $c_0 = \mathbf{u}^{\mathrm{T}}\mathbf{s} + x + b\lfloor q/2 \rfloor$;

g) обчислюється $\mathbf{c}_i = \mathbf{F}_{id}{}^{\mathrm{T}}\mathbf{s} + \begin{bmatrix} y \\ z \end{bmatrix} \in Z_q^{2m}$.

Зашифрованим текстом є пара $(c_0, \mathbf{c}_1)$.

4. Алгоритм розшифрування (Decrypt).

Розшифрування криптограми біта повідомлення $(c_0, \mathbf{c}_1)$ для ідентифікатора *id*, за наявності отриманого від центру секретного ключа $\mathbf{e}_{id}$ для даного ідентифікатора, виконується наступним чином:

a) обчислюється $w = c_0 - \mathbf{e}_{id}^{\mathrm{T}}\mathbf{c}_1 \in Z_q$;

b) як цілі числа порівнюються w і $\lfloor q/2 \rfloor$, якщо $|w - \lfloor q/2 \rfloor| < \lfloor q/4 \rfloor|$, вихідним значенням є 1; інакше – 0.

Відмітимо, що на повідомлення накладається маска аналогічно з попередньою схемою. Стандартний аналіз показує, що розшифрування є успішним з більш ніж переважною ймовірністю. Матриця **R** грає ключову роль в забезпеченні захищеності схеми. Безпека даної схеми доведена в селективно-ідентифікаційній моделі. Мета криптоаналітика – створення ідентифікатора користувача id^*, який потім встановлюється в IBE-схемі замість *id*.

3. АНАЛІЗ БЕЗПЕКИ СХЕМ, ЩО ЗАСНОВАНІ НА LWE-ЗАДАЧАХ

У реальній IBE-схемі, функція з секретом для матриці $\mathbf{A}_0$ є відомою. З іншого боку, для схем, заснованих на LWE-задачах, функція з секретом для матриці $\mathbf{A}_0$ є не відомою. Але центр повинен бути в змозі відповісти на запити вироблення ключа. Це є можливим за умови створення функції з секретом для матриці **B** і використання алгоритму SampleRight наступним чином.

Припустимо, що метою криптоаналітика є створення ідентифікатора користувача id^*. Центр встановлює IBE-схему, в звичайному порядку, генеруючи **u**. Далі центр генерує випадкову матрицю $\mathbf{A}_0 \in Z_q^{n \times m}$ і пару $(\mathbf{B}, \mathbf{T})$, використовуючи алгоритм генерації функції з секретом, де $\mathbf{B} \in Z_q^{n \times m}$ і **T** скорочений базис для решітки $\Lambda^{\perp}(\mathbf{B})$. Для генерування зашифрованого тексту необхідна випадкова матриця $\mathbf{R} \in \{-1,1\}^{m \times m}$. Так як ці параметри не залежать від запитів криптоаналітика, то вони обираються при налаштуваннях. Матриця $\mathbf{A}_1$ визначається, як:

$$\mathbf{A}_1 = \mathbf{A}_0, \mathbf{R}^* - H(id^*)\mathbf{B}.$$

Оголошуються відкриті параметри $(\mathbf{A}_0, \mathbf{A}_1, \mathbf{B}, \mathbf{u})$. При цьому центр не має функції з секретом для $\mathbf{A}_0$, але має функцію з секретом для **B**.

Припустимо, що криптоаналітик робить запит на вироблення ключа для ідентифікатора *id*. Тоді $\mathbf{F}_{id}$ формується наступним чином:

$$\mathbf{F}_{id} = [\mathbf{A}_0, \mathbf{A}_1 + H(id)\mathbf{B}] =$$
$$= [\mathbf{A}_0, \mathbf{A}_0\mathbf{R}^* + (H(id) - H(id^*))\mathbf{B}].$$

З властивості про повно рангову різницю H, витікає, що $H(id) - H(id^*)$ не сингулярне. Також, з великою ймовірністю **B** не сингулярне, і, тоді $\mathbf{B}' = (H(id) - H(id^*))\mathbf{B}$ також не сингулярне. Знання функції з секретом **T** для **B** дозволяє центру за допомогою алгоритму SampleRight обчислити прообраз **u** для $\mathbf{F}_{id}$. Причому, алгоритм SampleRight вимагає скорочений базис для **B'**, тоді, як насправді центр має скорочений базис для **B**. Таким чином, центр може відповісти на будь-які запити вироблення ключа, крім для id^*.

Розподілення $(c_0^*, \mathbf{c}_1^*)$ в реальних схемах є таким, що в LWE-випадку вхідні значення мають «реальне» розподілення. В останньому доказі безпеки, як є прийнятим, це розподілення є «випадковим», тобто, $(c_0^*, \mathbf{c}_1^*)$ обирається рівно ймовірно і випадково з $Z_q \times Z_q^{2m}$. Тоді криптоаналітик не має переваги в атаці селективної підробки ідентифікатора. Крім того, показано, що відмінності між цими двома атаками обмежені зверху перевагою рішення LWE-задачі.

4. НЕЧІТКЕ IBE-ШИФРУВАННЯ НА РЕШІТКАХ

З розвитком схеми шифрування, питання забезпечення підтримки комплексних політик доступу, стає все більш актуальним. Зокрема, стає необхідним поява систем, на основі функціонального шифрування. В даних схемах, власник секретного ключа може розшифрувати дані, та/або будь-яку частину, та/або функцію від цих даних, не просто на основі рішення одержувача (одержувачів), а на основі сформованої політики. Переваги таких схем є очевидними – доступ до зашифрованих даних виходить за рамки простого переліку, стаючи потенційно довільною функцією.

З моменту появи нечіткого шифрування на ідентифікаторах [9], з'явилися кілька систем, що вийшли за рамки традиційної «призначено отримувачу» парадигми шифрування. У рамках напрямку даних робіт [10, 11], ключ,або, в деяких варіантах, зашифрований текст, пов'язується з предикатом, скажімо, функцією f, в той же час, зашифрований текст (або ключ) пов'язаний з атрибутом, скажімо, вектором x. Розшифрування є успішним, тоді і тільки тоді, коли $f(x)=1$. Зокрема, шифрування на основі атрибутів [12-17], відноситься до випадку, коли предикат у вигляді булевої формули, з даними атрибутами, забезпечує двійкові входи. Нечітке IBE-шифрування являє собою особливий випадок, коли функція f є *k-out-of-l* пороговою функцією. У шифруванні, на основі предикатів, предикат f повинен обчислюватись без будь-яких знань про атрибути, окрім знань двійкової вихідної послідовності з $f(x)$. Однак, конструкції такого типу поки що обмежені предикатами, що задаються всередині схеми, тобто, впровадженими константами і атрибутами деякого поля.

Схема нечіткого IBE-шифрування на решітках

С. Агравал та ін. у роботі [18] запропонували таку схему. Нехай $\lambda \in Z^+$ – буде параметром безпеки. Нехай $q=q(\lambda)$ буде простим, $n=n(\lambda)$ і $m=m(\lambda)$ два позитивних цілих числа, $\sigma=\sigma(\lambda)$ і $\alpha=\alpha(\lambda)$ два позитивних параметра Гауса. Припустимо, що $\mathrm{id}\in\{0,1\}^l$, для деякого $l\in N$.

1. Встановлення параметрів (Set-Up)

Алгоритм за вхідні значення приймає параметр безпеки λ і довжину ідентифікатора l:

1. Використовується алгоритм **TrapGen**(1^λ) для вибору $2l$ рівномірних випадкових $n\times m$ матриць $A_{i,b}\in Z_q^{n\times m}$ (для усіх $i\in[l]$, $b\in\{0,1\}$) разом з повно-ранговим набором векторів $\mathbf{T}_{i,b}\subseteq\Lambda_q^{\perp}(\mathbf{A}_{i,b})$, таких, що $\|\tilde{\mathbf{T}}_{i,b}\|\le m\cdot w(\sqrt{\log m})$.

2. Обирається рівномірний і випадковий вектор $\mathbf{u}\in Z_q^n$.

Відкритими параметрами є:

$$\mathrm{PP}=(\{\mathbf{A}_{i,b}\}_{i\in[l],b\in\{0,1\}},\mathbf{u}).$$

Майстер ключ: $\mathrm{MK}=(\{\mathbf{T}_{i,b}\}_{i\in[l],b\in\{0,1\}})$.

2. Вироблення секретного ключа користувача (Key-Gen)

Алгоритм за вхідні значення приймає відкриті параметри PP, майстер ключ MK, ідентифікатор $\mathrm{id}\in\{0,1\}^l$ і поріг $k\le l$:

1. Будується l частин від $\mathbf{u}=(u_1,\ldots,u_n)\in Z_q^n$, шляхом застосування схеми розподілу секрету Шаміра незалежно для кожної координати $\mathbf{u}$. А саме, для кожного $j\in[n]$, обирається рівномірно і випадково поліном $p_j\in Z_q[x]$ ступеню $k-1$, такий, що $p_j(0)=u_j$.

Будується j - та частка вектора:

$$\hat{\mathbf{u}}_j=(\hat{u}_{j,1},\ldots,\hat{u}_{j,n})\overset{\mathrm{def}}{=}$$
$$=(p_1(j),p_2(j),\ldots,p_n(j))\in Z_q^n.$$

Забігаючи наперед (до розшифрування), відмітимо, що для всіх $J\subset[l]$, таких, що $|J|\ge k$, можна обчислити дробові коефіцієнти Лагранжа L_j, такі, що $\mathbf{u}=\sum_{j\in J}L_j\cdot\hat{\mathbf{u}}_j(\mathrm{mod}\,q)$. Тобто, ми інтерпретуємо L_j у вигляді дробі цілих чисел, яку також можна оцінити $(\mathrm{mod}\,q)$.

2. Використовуючи функцію з секретом MK і алгоритм SamplePre, знаходимо $\mathbf{e}_j\in Z^m$, таке, що $\mathbf{A}_{j,\mathrm{id}_j}\cdot\mathbf{e}_j=\hat{\mathbf{u}}_j$, для $j\in[l]$.

Секретним ключем для даного id є: $(\mathbf{e}_1,\ldots,\mathbf{e}_l)$.

3. Алгоритм зашифрування (Encrypt).

Алгоритм за вхідні значення має: відкриті параметри PP, ідентифікатор id, повідомлення $b\in\{0,1\}$:

1. Нехай $D\overset{def}{=}(l!)^2$.
2. Рівномірно та випадково обирається $\mathbf{s}\xleftarrow{R}Z_q^n$.
3. Обирається терм шуму $x\leftarrow\chi_{\{\alpha,q\}}$ і $\mathbf{x}_i\leftarrow\chi_{\{\alpha,q\}}{}^m$.
4. Встановлюється $c_0\leftarrow\mathbf{u}^{\mathrm{T}}\mathbf{s}+Dx+b\lfloor q/2\rfloor\in Z_q$.
5. Встановлюється $\mathbf{c}_i\leftarrow\mathbf{A}_{i,\mathrm{id}_i}{}^{\mathrm{T}}\mathbf{s}+D\mathbf{x}_i\in Z_q^m$ для всіх $i\in[l]$.

Зашифрованим текстом є: $\mathbf{CT}_{\mathrm{id}}:=(c_0,\{\mathbf{c}_i\}_{i\in[l]})$.

4. Алгоритм розшифрування (Decrypt)

Алгоритм на вході має відкриті параметри PP, секретний ключ $\mathrm{SK}_{\mathrm{id}}$ і зашифрований текст $\mathrm{CT}_{\mathrm{id}'}$:

1. Нехай $J\subset[l]$ позначає множину відповідних бітів для id та id′. Якщо $|J|<k$, на виході є $\perp$. В іншому випадку, можна обчислити дробові коефіцієнти Лагранжа L_j, так, що $\sum_{j\in J}L_j\mathbf{A}_j\mathbf{e}_j=\mathbf{u}(\mathrm{mod}\,q)$.

2. Обчислюється

$$r\leftarrow c_0-\textstyle\sum_{j\in J}L_j\cdot\mathbf{e}_j^{\mathrm{T}}\mathbf{c}_j(\mathrm{mod}\,q),$$

де $r\in\left[-\lfloor q/2\rfloor,\lfloor q/2\rfloor\right)\subset Z$.

3. Якщо $|r|<q/4$, вихідним значенням є 0, інакше 1.

5. АНАЛІЗ БЕЗПЕКИ, ОЦІНКА ВИБОРУ СИСТЕМНИХ ПАРАМЕТРІВ І ДОВЖИН КЛЮЧІВ ДЛЯ НЕЧІТКОГО IBE-ШИФРУВАННЯ НА РЕШІТКАХ

Аналіз безпеки нечітких IBE-схем на решітках

Вважається, що конструкція нечіткого IBE-шифрування забезпечує селективно-ідентифікаційну безпеку. При цьому, передбачається, що обраний зашифрований текст нічим не відрізняється від випадкового елемента в просторі зашифрованого тексту [19].

Твердження 1. Якщо існує PPT зловмисник A з перевагою $T>0$ в грі над селективною моделлю безпеки для нечіткої IBE-схеми з попереднього розділу, тоді існує PPT алгоритм B, який вирішує LWE-задачу з перевагою $T/(l+1)$.

Доведення твердження. LWE-задача, на прикладі, реалізується як оракул O, який може бути дійсно випадковим $O_\$$, або псевдовипадковим O_s оракулом шуму, для деякого секретного ключа $s \in Z_q^n$. Центр B використовується криптоаналітиком A для того, щоб розрізнити ці два оракула, тоді можна зімітувати алгоритм дій криптоаналітика. Нехай A повідомляє B ідентифікатор id^*, замінюючи ним дійсний id.

Центр B звертається до O, отримує $(lm+1)$ LWE вибірок, визначені, як:

$$\{(\mathbf{w}_1^1, v_1^1),(\mathbf{w}_1^2, v_1^2),\ldots,(\mathbf{w}_1^m, v_1^m)\},\ldots,$$
$$\{(\mathbf{w}_l^1, v_l^1),(\mathbf{w}_l^2, v_l^2),\ldots,(\mathbf{w}_l^m, v_l^m)\} \in \{Z_q^n \times Z_q\}^{(lm+1)}.$$

1. Встановлення параметрів. Центр B встановлює відкриті системні параметри PP, наступним чином:

a. Обирається l матриць $\mathbf{A}_{i,\mathrm{id}_i^*}$, $i \in [l]$ з LWE задачі $\{(\mathbf{w}_i^1),(\mathbf{w}_i^2),\ldots,(\mathbf{w}_i^m)\}_{i\in[l]}$. Обирається l матриць $\mathbf{A}_{i,\overline{id}_i^*}$, $i \in [l]$, використовуючи TrapGen з функцією з секретом $\mathbf{T}_{i,\overline{id}_i^*}$.

b. Будується вектор $\mathbf{u}$, з LWE задачі, $\mathbf{u} = \mathbf{w}_1$.

Відкриті параметри стають доступними зловмиснику.

2. Вироблення секретного ключа. Центр B відповідає на кожен запит вироблення секретного ключа для ідентифікатора id наступним чином:

a. Нехай $id \cap id^* := I \subset [l]$ і нехай $|I| = t < k$. Тоді, відмітимо, що B має функції з секретом для матриць відповідного набору $\overline{I}$, де $|\overline{I}| = l - t$. Припустимо, що перші t біти id дорівнюють id^*.

b. Символічно представимо частини $\mathbf{u}$, як $\hat{\mathbf{u}}_i = \mathbf{u} + \mathbf{a}_1 i + \mathbf{a}_2 i^2 + \ldots + \mathbf{a}_{k-1} i^{k-1}$, де кожен $\mathbf{a}_1,\ldots,\mathbf{a}_{k-1}$ є вектором змінної довжини n.

c. Для i, $id_i^* = id_i$, випадково обирається $\mathbf{e}_i$, використовуючи алгоритм SampleGaussian. Встановлюється $\mathbf{A}_{i,\mathrm{id}_i}\mathbf{e}_i$; $i \in [t]$.

d. Доки $t \le k-1$, і $k-1$ змінюється $\mathbf{a}_1,\ldots,\mathbf{a}_{k-1}$, випадково обирається $k-1-t$ частин з $\hat{\mathbf{u}}_{t+1},\ldots,\hat{\mathbf{u}}_{k-1}$, значення $\mathbf{a}_1,\ldots,\mathbf{a}_{k-1}$ є визначеними. Таким чином визначаються всі l частини $\hat{\mathbf{u}}_1,\ldots,\hat{\mathbf{u}}_l$.

e. Для знаходження $\mathbf{e}_j$, такі, що $\mathbf{A}_{j,\mathrm{id}_j}\mathbf{e}_j = \hat{\mathbf{u}}_j$, для $j = t+1,\ldots l$ застосовується алгоритм

$\mathbf{SamplePre}(\mathbf{A}_{j,\mathrm{id}_j}, \mathbf{T}_{j,\mathrm{id}_j}, \hat{\mathbf{u}}_j, \sigma)$.

f. Повертається $(\mathbf{e}_1,\ldots,\mathbf{e}_l)$.

Відмітимо, що розподіл відкритих параметрів і ключів в реальній схемі статистично не відрізняється від даної імітації.

3. Зашифрування повідомлення. A на виході має бітове повідомлення $b^* \in \{0, 1\}$. B на запит формує зашифрований текст для id^*:

a. Нехай $c_0 = Dv_1 + b\lfloor q/2 \rfloor$.

b. Нехай $c_i = (Dv_i^1, Dv_i^2, \ldots, Dv_i^m)$ для $i \in [l]$.

4. Розшифрування повідомлення. Зловмисник у якості вихідних даних має приблизне значення b'. Імітатор B використовує це припущення для визначення відповіді LWE оракула: якщо $b' = b^*$, тоді вихідне значення є «справжнім», інакше вихідне значення є «випадковим».

Вибір параметрів для нечітких IBE-схем на решітках

Для того, щоб результат дешифрування був коректним, та для забезпечення необхідного рівня захищеності, параметр безпеки n, верхня межа l, розмір множини та інші параметри, задаються при наступних обмеженнях [20]:

1. Для алгоритму генерування решітки з функцією з секретом необхідно, щоб $m \ge 5n\log q$. Враховуючи це обмеження для m, вихідною послідовністю алгоритму TrapGen є базис Грама-Шміта довжиною не більше $m \cdot \sqrt{\log m}$. Використовуючи алгоритм SamplePre, секретний ключ вектор $\mathbf{e}_j$, є взятим з дискретного рівняння Гауса зі стандартним відхиленням $\sigma \ge m \cdot \log m$, і, таким чином, з майже експоненціально малої ймовірністю, має довжину не більшу за $\sigma\sqrt{m} \le m^{1.5} \cdot \log m$.

2. Встановлюється розподілення з шумом $\chi = \bar{\Psi}_\alpha^m$, де $\alpha \ge 2\sqrt{m}/q$, до якого застосовується редукція Регева [19]. Вектор $\mathbf{x}$, вибраний з цього розподілення, має довжину $O(\alpha q\sqrt{m}) \le 2m$ з майже експоненціально малої ймовірністю.

3. Для коректності, необхідно, щоб задовольнялося рівняння:

$$|Dx - \sum_{j\in J} DL_j \mathbf{e}_j^{\mathrm{T}} \mathbf{x}_j| \le D|x| + \sum_{j\in J} D^2 |\mathbf{e}_j^{\mathrm{T}} \mathbf{x}_j| < q/4.$$

Оскільки $D = (l!)^2$, маємо:

$$|D|x| + \sum_{j\in J} D^2 |\mathbf{e}_j^{\mathrm{T}} \mathbf{x}_j| \le$$
$$\le D \cdot \alpha q\sqrt{m} + l \cdot D^2 \cdot (\alpha q\sqrt{m} \cdot m^{1.5} \log m \cdot \sqrt{m}) \le,$$
$$\le 4 \cdot m^3 \log m \cdot l(l!)^4 \le m^3 \log m \cdot 2^{5l}$$

причому, $(l!)^4 \le (l!)^{4l} \le 2^{5l}$. Параметр $q \ge m^3 \log m \cdot 2^{5l}$ забезпечує коректність.

Стосовно налаштування конкретних параметрів під ці обмеження, враховуючи постійну $\epsilon \in (0, 1)$, встановлюється:

- Параметр безпеки $n = l^{1/\epsilon}$.
- Модуль q повинен бути простим числом в інтервалі $[n^6 2^{5l}, 2 \cdot n^6 2^{5l}]$.
- $m = n^{1.5} \ge 5n\log q$, повинно задовольняти (1).

Проаналізувавши останні два пункти, можна побачити, що $q \ge m^3 \log m \cdot 2^{5l}$, а параметр шуму $\alpha = 2\sqrt{m}/q = 1/(2^{5n\epsilon} \cdot \mathrm{poly}(n))$.

Зв'язавши наведене вище, в найгіршому та середньому випадку, отримуємо захищеність, що відповідає стійкості $2^{O(n^\epsilon)}$-апроксимацій gapSVP або SIVP на n- мірних решітках, якщо використовувати алгоритми, що працюють в часі $q \cdot \mathrm{poly}(n) = 2^{O(n^\epsilon)}$. З даним рівнем знань алгоритм є LWE- стійким для $\epsilon < 1/2$.

6. ПОЄДНАННЯ МАТЕМАТИКИ АЛГЕБРАЇЧНИХ РЕШІТОК ЗІ СПАРЮВАННЯМИ ТОЧОК ЕК

Флоріан Гесс у роботі [21] запропонував математичний апарат, що поєднує математику спарювань точок ЕК та алгебраїчні решітки.

Для кільця A з ідеалом $I^{(i)}$ нехай, для зручності обчислень: $R=Z$ і $R=Q[t]$. Нехай R – домен (головний ідеал), і нехай $r,s\in R$, для $r\neq 0$, не рівних одиниці, і для s, що мають порядок $n\geq 2$ в $(R/rR)^{\times}$. Іншими словами, s – первісний корінь n-го ступеню з одиниці по модулю r.

Визначимо R-алгебру та її ідеали:

$$A=R[x]/(x^n-1)R[x],$$
$$I^{(i)}=\{h+(x^n-1)R[x]\,|\,h(s)\equiv 0 \bmod r^iR\}, \quad (1)$$

для $i\geq 0$, таких, що $s^n\equiv 1 \bmod r^iR$. Надалі елементи будуть ототожнюватися з їх поліноміальним представленням ступеню $\leq n-1$. Визначимо також R- модулі:

$$I^{(i),m}=\{h\in I^{(i)}\,|\,\deg(h)\leq m-1\}.$$

Відмітимо, що $I^{(i),m}\subseteq I^{(j),w}$, для $m\leq w$ і $j\leq i$, а також $I^{(i),n}=I^{(i)}$.

Структура ідеалу

Лема 1. *Ідеали $I^{(i)}$ та $I^{(i),m}$ мають наступні властивості:*

1. $I^{(i)}=r^iA+(x-s)A$.
2. $I^{(i),m}$ має вільний ранг m та базис $r^i, x-s, x^2-s^2,\ldots,x^{m-1}-s^{m-1}$.
3. Якщо $m\geq\phi(n)$, тоді $I^{(i),m}=M\oplus I^{(i),\phi(n)}$ з $M=\{h\in I^{(i),m}\,|\,h\equiv 0 \bmod \Phi_n\}$.

Доведення. З визначення $I^{(i)}$ ясно, що $r^iA+(x-s)A\subseteq I^{(i)}$. З іншого боку, нехай $h\in I^{(i)}$. Поліноміальний поділ $x-s$ із залишком показує, що $h=g\cdot(x-s)+h(s)$ з $g\in A$ і $h(s)\in R$. З визначення $I^{(i)}$ маємо $h(s)\in r^iR$. Отже, $h=h(s)+g\cdot(x-s)\in r^iA+(x-s)A$. Перше твердження доведене.

Друге твердження виводиться з першого та із короткої нормальної форми Ерміта, що обчислена над базисом $r^i, x-s, x(x-s),\ldots,x^{m-2}(x-s)$ для $I^{(i),m}$.

Трете твердження виводиться за допомогою поліноміального поділу Φ_n із залишком. Відображення $I^{(i),m}\to I^{(i),\phi(n)}$, $h\mapsto h \bmod \Phi_n$ ділиться на включення $I^{(i),\phi(n)}\to I^{(i),m}$. Де $\Phi_n\in I^{(i),\phi(n)}$, доки $\Phi_n(s)\equiv 0 \bmod r^i$. Відмітимо, що M – це вільний R-модуль з базисом $\Phi_n,\ldots,x^{m-\phi(n)-1}\Phi_n$.

Зазначимо, що додатково до Леми 1, можна показати, що $I^{(i)}=(I^{(1)})^i$, якщо $R=nR+rR$ (наприклад, якщо $R=Z$ і r – просте). Так як ідеал $I^{(i)}$ є замкнутим щодо множення на x, можна побачити, що він є замкнутим при обертанні коефіцієнтів $h\in I^{(i)}$.

Аргументи для решітки з $R=Z$

Нехай $R=Z$ та $r\geq 2$. Для $h=\sum_{i=0}^{d}h_ix^i\in Z[x]$, визначимо:

$$\|h\|_1=\sum_{i=0}^{d}|h_i| \quad\text{та}\quad \|h\|_2=\left(\sum_{i=0}^{d}|h_i|^2\right)^{1/2}.$$

Поширимо це визначення для A за допомогою представників класу ступенів $\leq n-1$. Це дозволить використовувати $I^{(i)}$ в математичних решітках. Маємо $\|\cdot\|_1=\Theta(\|\cdot\|_2)$ для $I^{(i)}$, де константи залежать тільки від n.

Лема 2. *Припустимо, що $i\geq 1$ задовольняє $s^n\equiv 1 \bmod r^i$ та нехай $h\in Z[x]$, так, що $h(s)\equiv 0 \bmod r^i$. Якщо $h\not\equiv 0 \bmod \Phi_n$, тоді:*

$$\|h\|_1\geq r^{i/\phi(n)}.$$

Доведення. Нехай ς – первісний корінь n- го ступеню з одиниці в $\bar{Q}$ і $B=Z[\varsigma]$ – кільце цілих чисел n-го ступеню поля циклотомічних чисел K/Q. Нехай $\alpha=r^iB+(\varsigma-s)B$. Тоді α – ідеал B з нормою $N_{K/Q}(\alpha)=r^i$, з припущенням по s. Маємо $\varsigma\equiv s \bmod \alpha$. Таким чином $h(\varsigma)\in\alpha\setminus\{0\}$ з припущенням по h, та, відповідно,

$$|N_{K/Q}(h(\varsigma))|\geq N_{K/Q}(\alpha)=r^i.$$

З іншого боку, $\phi(n)$ комплексно зв'язує $\varsigma^{(i)}$ з ς задовольняючи $|\varsigma^{(i)}|=1$. Тоді $|h(\varsigma^{(i)})|\leq\|h\|_1$ та:

$$|N_{K/Q}(h(\varsigma))|=|\prod_{j=1}^{\phi(n)}h(\varsigma^{(j)})|\leq\|h\|_1^{\phi(n)}$$

Об'єднання двох нерівностей доводить перше твердження.

Лема 3. *Припустимо, що $s^n\equiv 1 \bmod r^2$. Нехай $m\geq\phi(n)$ і $w=m-\phi(n)$. Будь-який LLL- зведений базис $v_1,\ldots,v_m$ над $I^{(1),m}$, з впорядкованої довжиною, задовольняє:*

$\|v_i\|_1=O(1)$ та $v_i\in I^{(2)}$, для $1\leq i\leq w$,

$\|v_i\|_1=\Theta(r^{1/\phi(n)})$ та $v_i\notin I^{(2)}$, для $w\leq i\leq m$.

Константи O і Θ залежать тільки від n і елемент, що співвідноситься з r, є достатньо великим порівняно з n.

Доведення. Згідно з Лемою 1, детермінантом $I^{(1),m}$ є r, з розмірністю m. Також, маємо $I^{(1),m}=M\oplus I^{(1),\phi(n)}$ з $M=\{h\in I^{(1),m}\,|\,h\equiv 0 \bmod \Phi_n\}$. Таким чином, існує хоча б $\phi(n)$ базисів векторів v_i над $I^{(1),m}$, що мають не нульові проекції на $I^{(1),\phi(n)}$. Згідно з Лемою 2, v_i задовольняють $\|v_i\|_2=\Omega(r^{1/\phi(n)})$. З іншого боку, LLL-властивість показує, що $\prod_{i=1}^{m}\|v_i\|_2=O(r)$. Таким чином, достеменно існує $\phi(n)$ базисів векторів v_i, з розміром $\Theta(r^{1/\phi(n)})$, що мають не нульові проекції на $I^{(1),\phi(n)}$. Інший базис векторів v_i в M та задовольняє $\|v_i\|_2=O(1)$. Оскільки v має задану довжину, звідси випливає і вибір норми.

Наразі $\Phi_n(s)\equiv 0 \bmod r^2$ за припущенням по s. Отже $v\in I^{(2)}$ для будь якого $v\in M$. Тоді, $v_i\in I^{(2)}$ для $1\leq i\leq w$. З іншого боку, якщо $v\in I^{(1),m}\setminus M$ та $v\in I^{(2)}$, тоді $v\not\equiv 0 \bmod {}_n^{\mathrm{l}}$ та $v(s)\equiv 0 \bmod r^2$. Тоді, згідно з Лемою 2, $\|v\|_2=\Omega(r^{2/\phi(n)})$, чого не може бути. Звідси, $v_i\notin I^{(2)}$ для $w\leq i\leq m$.

Дійсні константи O-термів і Θ-термів є важко обчислюваними, та є доступними тільки за умови обмежень гіршого випадку, які, зазвичай, занадто великі. Оскільки на практиці r є значно більшим за n, вплив цього терму є настільки малим, що ним можна знехтувати. У цьому випадку залежності елемента зберігаються. Відмітимо, що будь-який LLL-зведений базис $I^{(1),m}$ повинен містити хоча б один базисний елемент, що не належить до $I^{(2)}$.

Аргументи для решітки з $R=Q[t]$

Нехай $R=Q[t]$ і $\deg(r)\ge 1$.

Для $h=\sum_{i=0}^{d} h_i x^i \in Q[t,x]$, з $h_i = Q[t]$ визначимо:

$$\deg_t h = \max_{0\le i\le d} \deg(h_i).$$

Поширимо це визначення для A за допомогою представників класу x ступенів $\le n-1$. Це дозволить використовувати $I^{(i)}$ в математичних решітках зі ступенем (мається на увазі, що $I^{(i)}$ – вільний $Q[t]$-модуль кінцевого рангу, підмножини якого обмежені значеннями ступенів кінцевовимірного простору векторів).

Лема 4. *Припустимо, що $i\ge 1$ задовольняє $s^n \equiv 1 \bmod r^i Q[t]$ та нехай $h\in Q[t,x]$, так, що $h(s)\equiv 0 \bmod r^i Q[t]$. Якщо $h\not\equiv 0 \bmod \Phi_n(x) Q[t,x]$, тоді:*

$$\deg_t(h) \ge i/\phi(n)\deg(r).$$

Доведення. Нехай ς – первісний корінь n-го ступеню з одиниці в $\bar{Q}$ і $B=Q[t,\varsigma]$ – ціле замикання $Q[t]$ в функціональному полі $K=Q(t,\varsigma)/Q$. Нехай $\alpha = r^i B + (\varsigma - s)B$. Тоді α – ідеал B з нормою $N_{K/Q}(\alpha)=r^i$, з припущенням по s. Маємо $\varsigma \equiv s \bmod \alpha$. Таким чином $h(\varsigma)\in\alpha$ з припущенням по h, та, відповідно:

$$\begin{aligned}\deg(N_{K/Q(t)}(h(\varsigma))) &\ge \deg(N_{K/Q(t)}(\alpha)) = \\ &= i\deg(r)\end{aligned} \quad (1)$$

З іншого боку, $\phi(n)$ розкладання Пюїзе в ряд ς за ступенем оцінки $Q(t)$ є майже сталою (часто без ненульових ступенів t), комплексно пов'язаних $\varsigma^{(i)}$ з ς, і тим самим задовольняє $\deg(\varsigma^{(i)})=0$. Тоді $\deg(h(\varsigma^{(i)})) \le \deg_t(h)$ та:

$$\deg(N_{K/Q}(h(\varsigma))) = \deg\left(\prod_{j=1}^{\phi(n)} h(\varsigma^{(j)})\right) =$$

$$= \prod_{j=1}^{\phi(n)} \deg(h(\varsigma^{(j)})) \le \phi(n)\deg_t(h).$$

Об'єднання двох нерівностей доводить дане твердження.

Наступна Лема використовує функціональні поля LLL. Вхідними значення $M\in Q[t]^{n\times n}$ з $\det(M)\ne 0$ функціонального поля LLL виходів $N,T\in Q[t]^{n\times n}$, такі, що $N=MT, \det(T)=1$ та сума максимальних ступенів в кожному стовпці $\det(M)$. Стовпці N є за визначенням LLL-зведеними елементами $Q[t]^n$.

Лема 5. *Припустимо, що $s^n \equiv 1 \bmod r^2 Q[t]$. Нехай $m\ge\phi(n)$ та $w=m-\phi(n)$. Будь-який упорядкований по довжині LLL-зведений базис $v_1,\dots,v_m$ над $I^{(1),m}$, задовольняє:*

$\deg_t v_i = 0$ та $v_i \in I^{(2)}$, для $1\le i\le w$,

$\deg_t(v_i) = 1/\phi(n)\deg(r)$ та $v_i \notin I^{(2)}$, для $w\le i\le m$.

Доведення. Доведення даної леми є в точності аналогічним доведенню Леми 3 (за аналогію використовується $\deg_t = (\|\cdot\|_2)$).

Спарювання на решітках

Нехай V_n – мультиплікативна група, вигляду:

$$V_n = \{wf \mid w\in F_q \cap \mu_{lcm(2,n)}, \\ f: G_1\times G_2 \to \mu_r\}$$

де G_1 та G_2 – циклічні групи простого порядку r та $\gcd(n,r)=1$. Нехай W_n позначає фактор групи V_n, обчислений шляхом факторизації константних функцій зі значеннями в $F_q\cap\mu_{\mathrm{lcm}(2,n)}$. Тоді елементами W_n є функції $G_1\times G_2\to\mu_r$, які визначені з точністю до скалярних кратних з $F_q\cap\mu_{\mathrm{lcm}(2,n)}$. Нехай W_n^{bilin} позначає підгрупу W_n, що генерована шляхом використання білінійних функцій.

Теорема 1. *Нехай r – просте число, взаємно просте з n, та s – примітивний корінь n-го ступеню з одиниці по модулю r^2. Нехай:*

$$a_s: I^{(1)} \to W_n, \quad h \mapsto a_{s,h}$$

буде відображенням, з наступними властивостями:

1. $a_{s,g+h} = a_{s,g}a_{s,h}$ *для всіх* $g,h\in I^{(1)}$;
2. $a_{s,hx} = a_{s,h}^s$ *для всіх* $h\in I^{(1)}$ *з* $a_{s,h}\in W_n^{bilin}$;
3. $a_{s,r}\in W_n^{bilin}\setminus\{1\}$ *та* $a_{s,t-s}=1$.

Тоді $\mathrm{im}(a_s)\in W_n^{bilin}$ *та* $\ker(a_s)=I^{(2)}$. *Точніше,* $a_{s,h}=a_{s,r}^{h(s)/r}$, *для всіх* $h\in I^{(1)}$.

Існує ефективно обчислюване $h\in I^{(1),\phi(n)}$ *з* $\|h\|_1 \ge O(r^{1/\phi(n)})$ *та* $a_{s,h}\ne 1$. *Константа O залежить тільки від n.*

Будь-яке $h\in I^{(1)}$ *з* $a_{s,h}\ne 1$ *задовольняє* $\|h\|_1 \ge r^{1/\phi(n)}$.

Доведення. З першої та другої властивостей можна побачити, що $a_{s,hg}=a_{s,h}^{g(s)}$ для всіх $h\in I^{(1)}$ з $a_{s,h}\in W_n^{bilin}$ і $g\in A$. З Леми 1, маємо $I^{(1)} = rA + (x-s)A$, отже, кожне $h\in I^{(1)}$ має вигляд $h=g_1 r + g_2(x-s)$ з $g_1,g_2\in A$. Тоді, використовуючи третю властивість можна обчислити:

$$a_{s,h} = a_{s,g_1 r + g_2(x-s)} = \\ = a_{s,r}^{g_1(s)} a_{s,x-s}^{g_2(s)} = a_{s,r}^{g_1(s)} \in W_n^{bilin}$$

і, отже, $\mathrm{im}(a_s)\subseteq W_n^{bilin}$. Доки $a_{s,r}\ne 1$ та r – просте, маємо $\mathrm{im}(a_s)\in W_n^{bilin}$.

Властивості a_s показують до яких пір можна спростити вираз. Беремо W_n^{bilin} за A-модулем,

через $f^g = f^{g(s)}$ для $f \in W_n^{bilin}$ та $g \in A$. Тоді a_s – епіморфізм з A-модулями $I^{(1)}$ і W_n^{bilin}.

Ядром a_s є A-подмодуль $I^{(1)}$ і, отже, ідеал A міститься в $I^{(1)}$. Оскільки a_s сюр'єктивно, індекс задовольняє $(I^{(1)} : \ker(a_s)) = \#W_n^{bilin} = r$. Але $r^2, x - s \in \ker(a_s)$, тоді за Лемою 1 $I^{(2)} = r^2 A + (x - s)A \subseteq \ker(a_s)$. Також з Леми 1 маємо $(I^{(1)} : I^{(2)}) = r$, отже $\ker(a_s) = I^{(2)}$.

З формули (1.1) можна побачити, що $g_1(s) = h(s)/r \bmod r$, і, отже $a_{s,h} = a_{s,r}^{h(s)/r}$, що показує зв'язок $a_{s,h}$ з генератором $a_{s,r} \in W_n^{bilin}$.

Використовуючи $\ker(a_s) = I^{(2)}$, інша частина теореми безпосередньо випливає з Леми 3 (враховуючи $m = \varphi(n)$), LLL-алгоритму і Леми 2.

Ідеал $I^{(1)}$ разом з відображенням $a_s : I^{(1)} \to W_n$, що задовольняє властивостям наведеним в Теоремі 1, називають спарюванням на решітці зі функцією спарювання решітки a_s.

Теорема 2. *Нехай $n \geq 2$ та r, s – змінні поліноми в $Z[t]$, такі, що s – примітивний корінь n-го ступеню з одиниці по модулю r^2, а r – примітивний поліном. припустимо також, що існує функція спарювання решітки:*

$$a_{s(t_0)} : I^{(1)}_{r(t_0),s(t_0)} \to W^{bilin}_{n,r(t_0)},$$

для всіх $t_0 \in J$, де J – відповідна необмежена підмножина Z.

Тоді існує $h \in Z[t][x]$ з $\deg(h) \leq \phi(n) - 1$ та $\deg_t(h) = 1/\phi(n)\deg(r)$, таке, що: $a_{s(t_0),h(t_0,x)} \neq 1$, для всіх досить великих $t_0 \in J$. Поліном h можна ефективно обчислити.

Будь-який $h \in Z[t][x]$, такий, що $a_{s(t_0),h(t_0,x)} \neq 1$, для всіх досить великих $t_0 \in J$, задовольняє умові $\deg_t(h) \geq 1/\phi(n)\deg(r)$.

Доведення. Існує лише кінцеве число $t_0 \in J$, таких, що $s(t_0)$, що мають порядок менший за $n \bmod r^2$, так як t_0 повинні бути нулями за $s^m - 1 \bmod r$ для $m < n$. Доки t_0 обирається досить великим, можна вважати, що $s(t_0)$ – примітивний корінь n-го ступеню з одиниці по модулю r^2.

Відповідно до початку цього розділу, визначимо $A, I^{(1)}, I^{(2)}$ для r, s і $R = Q[t]$. З властивості Леми 5 $m = \varphi(n)$ і функції LLL- поля, можна побачити, що існують $v_i \in Q[t][x]$ з $v_i(s) \equiv 0 \bmod rQ[t]$, $\deg(v_i) \leq \varphi(n) - 1$ та $\deg_t(v_i) = 1/\phi(n)\deg(r)$ для $1 \leq i \leq \phi(n)$. Нехай $h \in Z[t][x]$ буде добутком v_i з найменшим спільним кратним всіх знаменників усіх Q-коефіцієнтів v_i. Тоді з Леми Гауса, $h(s) \in Z[t]$ та $h(s) \equiv 0 \bmod rZ[t]$, оскільки r вважається примітивним.

Підставляючи в порівняння t_0 замість t, маємо $h(t_0, s(t_0)) \equiv 0 \bmod r(t_0)$. З $\deg_t(h) = 1/\phi(n)\deg(r)$ можна побачити $\|h(t_0, x)\|_1 = O(r(t_0)^{1/\phi(n)})$. З Леми 2 випливає, що $h(t_0, s(t_0)) \not\equiv 0 \bmod r(t_0)^2$. Робимо висновок, що $a_{s(t_0),h(t_0,x)}$ визначає не вироджене спарювання.

Останнє твердження про ступінь випливає з $\|h(t_0, x)\|_1 \geq r(t_0)^{1/\phi(n)}$ за Лемою 2, якщо t_0 прагне до нескінченності.

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

Більшість існуючих схем, що базуються на ідентифікаторах, використовують математику білінійних відображень точок на еліптичних кривих. Складність даних перетворень базується на вирішенні задачі дискретного логарифму в групі точок еліптичної кривої, та знаходься в межах між субекспоненційною і експоненційною. Криптографічні алгоритми, що базуються на спарюваннях, не мають прийнятної швидкодії.

Альтернативою математиці спарюванням точок є математика, що використовує перетворення в кільцях зрізаних поліномів – алгебраїчні решітки. Дані обчислення мають лінійну складність обчислень, яка дорівнює $O(n2)$, що забезпечує швидкодію обчислень. Криптографічна стійкість таких алгоритмів ґрунтується на складності вирішенні задачі знаходження найкоротшого вектора в заданій решітці. На даний момент не існує алгоритмів, які б знаходили найкоротший вектор зі складністю меншу за експоненційну.

Таким чином, до основних переваг криптосистем на решітках можна віднести швидкодію, що є наближеною до швидкодії симетричних алгоритмів, за умови використання розпаралелювання, а також стійкість до атак з використанням навіть квантових обчислень. Але дані системи мають і недоліки, основним з яких є занадто великий розмір ключів та зашифрованого тексту.

Проаналізувавши основні IBE-схеми, що використовують алгебраїчні решітки, можна відмітити, що:

1. Схеми на основі функцій з секретом, з обраним прообразом, тобто з підібраним базисом за допомогою розподілення Гауса, зі стандартним відхиленням, що наближається до найбільшого з векторів Грама-Шмита, отриманих завдяки ортогоналізації цього базису, забезпечують складність найгіршого випадку вирішення задачі на решітках, в моделі випадкового оракулу.

2. Складність схем, що базуються на вирішенні LWE-задачі, заснована найгіршому випадку квантової складності наближеної SVP-задачі. Що означає, що розкриття таких схем тягне за собою появу ефективних квантових алгоритмів вирішення наближеної SVP-задачі, що є малоймовірним. Квантова обчислювальна модель використовується тільки при зведенні задач теорії решіток до LWE-задач, сама LWE-задача, так само, як і всі системи шифрування засновані на ній, використовує класичну обчислювальну модель.

3. Системи шифрування засновані на LWE-задачах мають деяку невелику ймовірність помилки дешифрування повідомлення, що зменшується, завдяки правильно підібраним параметрам (див. пункт 5). Також, цю ймовірність можна зменшити

до несуттєвої, завдяки використанню завадостійкого кодування до етапу зашифрування.

4. Алгоритми шифрування засновані на LWE- задачах мають наступні властивості:

- розміри секретного ключа $nl\log q$;
- розмір відкритого ключа $m(n+l)\log q$;
- розмір повідомлення $l\log t$;
- розмір зашифрованого тексту $(n+l)\log q$;
- зашифрування веде до росту повідомлення у $(1+\frac{n}{l})\log q / \log t$ разів;
- для зашифрування одного біту повідомлення необхідно $\tilde{O}(m(1+\frac{n}{l}))$ операцій;
- для розшифрування одного біту повідомлення необхідно $\tilde{O}(n)$ операцій.

В подальшому повинні бути вирішеними такі проблемні питання і задачі.

1. Теоретичні дослідження існуючих криптографічних алгоритмів на ідентифікаторах, що використовують математичні решітки.

2. Обґрунтування та побудування захищеної криптографічної схеми на основі математики алгебраїчних решіток.

3. Удосконалення IBE-систем шифрування на решітках, зокрема методів оптимізації довжини ключового матеріалу, за допомогою використання структурних циклічних решіток.

4. Зіставлення алгоритмів шифрування на решітках з алгоритмами шифрування на основі теоретики-числових проблем. Так, якщо існує оракул, що вирішує $\sqrt{n}$ -наближені SVP-задачі факторизації цілого числа, або дискретного логарифму, тоді алгоритми на решітках є більш крипто стійкими, ніж алгоритми на основі теоретики-числових проблем.

5. Пошук методів та засобів оцінки, обґрунтування критеріїв IBE-алгоритмів, що використовують алгебраїчні решітки.

6. Подальші дослідження стосовно можливості побудування алгоритмів на основі поєднання математик алгебраїчних решіток і спарювань точок еліптичних кривих.

Література

[1] *Clifford Cocks.* An identity based encryption scheme based on quadratic residues / In Bahram Honary edit. // IMA Int. Conf. «Lecture Notes in Computer Science». – Vol.2260. – Springer. – 2001. – P.360–363.

[2] *Dan Boneh.* Space-efficient identity based encryption without pairings / Dan Boneh, Craig Gentry, Michael Hamburg // IEEE Computer Society, FOCS. – 2007. – P.647–657.

[3] *Craig Gentry.* Trapdoors for hard lattices and new cryptographic constructions / Craig Gentry, Chris Peikert, Vinod Vaikuntanathan // In Richard E. Ladner and Cynthia Dwork edit., STOC, ACM. – 2008. – P.197–206.

[4] *Shweta Agrawal.* Efficient lattice (H)IBE in the standard model / Shweta Agrawal, Dan Boneh, Xavier Boyen // In Henri Gilbert edit. – EUROCRYPT Int. Conf. «Lecture Notes in Computer Science». – Vol.6110. – Springer. – 2010. – P.553-572.

[5] *Dan Boneh, Matthew K. Franklin.* Identity-based encryp tion from the Weil pairing / Dan Boneh, Matthew K. Franklin // SIAM Journal on Computing. – Vol. 32(3). – 2003. – P.586-615.

[6] *Chris Peikert.* Bonsai trees (or, arboriculture in lattice-based cryptography) [Електронний ресурс] / Chris Peikert // Cryptology ePrint Archive. – Report 2009/359. – 2009. – Режим доступу: http://eprint.iacr.org/.

[7] *Shweta Agrawal.* Lattice basis delegation in fixed dimension and shorter -ciphertext hierarchical IBE / Shweta Agrawal, Dan Boneh, Xavier Boyen // In Tal Rabin edit. – CRYPTO «Lecture Notes in Computer Science». – Vol.6223. – Springer. – 2010. – P.98-115.

[8] *Miklos Ajtai.* Generating hard instances of the short basis problem / Miklos Ajtai // In Jirэ Wiedermann, Peter van Emde Boas, Mogens Nielsen edit. – ICALP «Lecture Notes in Computer Science». – Vol.1644. – Springer. – 1999. – P.1-9.

[9] *Amit Sahai, Brent Waters.* Fuzzy identity-based encryption / Amit Sahai, Brent Waters // EUROCRYPT. – 2005. – P.457-473.

[10] *Jonathan Katz.* Predicate encryption supporting disjunctions, polynomial equations, and inner products / Jonathan Katz, Amit Sahai, Brent Waters // EUROCRYPT'08 «Proceedings of the theory and applications of cryptographic techniques 27th annual international conference on Advances in cryptology» – Springer-VerlagBerlin, Heidelberg – 2008. – P.146-162.

[11] *Allison B. Lewko.* Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption / Allison B. Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, Brent Waters // EUROCRYPT. – 2010. – P.62-91.

[12] *John Bethencourt.* Ciphertext-policy attribute-based encryption / John Bethencourt, Amit Sahai, Brent Waters // In SP '07 «Proceedings of the 2007 IEEE Symposium on Security and Privacy». – IEEE Computer Society. – Washington, DC, USA. – 2007. – P.321-334.

[13] *Ling Cheung, Calvin C. Newport.* Provably secure ciphertext policy ABE / Ling Cheung, Calvin C. Newport // ACM conference Computer and Communications Security. – 2007. – P.456-465.

[14] *Vipul Goyal.* Attribute-based encryption for fine-grained access control of encrypted data / Vipul Goyal, Omkant Pandey, Amit Sahai, Brent Waters // CCS '06 «Proceedings of the 13th ACM conference on Computer and communications security». – ACM. –New York, USA – 2006. – P.89–98.

[15] *Allison B. Lewko.* Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption / Allison B. Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, Brent Waters // EUROCRYPT. – 2010. – P.62-91.

[16] *Allison B. Lewko, Brent Waters.* Unbounded HIBE and attribute-based encryption / Allison B. Lewko, Brent Waters // EUROCRYPT. – 2011. – P.547-567.

[17] *Rafail Ostrovsky.* Attribute-based encryption with non-monotonic access structures / Rafail Ostrovsky, Amit Sahai, Brent Waters // CCS '07 «Proceedings of the 14th ACM conference on Computer and communications security». – ACM. –New York, USA – 2007. – P.195–203.

[18] *Shweta Agrawal*. Fuzzy Identity Based Encryption from Lattices [Електронний ресурс] / Shweta Agrawal, Xavier Boyen, Vinod Vaikuntanathan, Panagiotis Voulgaris, Hoeteck Wee // Cryptology ePrint Archive. – Report 2011/414. – 2011. – Режим доступу: http://eprint.iacr.org/.

[19] *Daniele Micciancio, Oded Regev*. Worst-case to average-case reductions based on Gaussian measures / Daniele Micciancio, Oded Regev // FOCS '04 «Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science». – IEEE Computer Society. – Washington, DC, USA. – 2004. – P.372-381.

[20] *Sanjit Chatterjee, Palash Sarkar*. Identity-Based Encryption / Sanjit Chatterjee, Palash Sarkar. // Springer Science + Business Media, LLC. – 2011. – P.125-135.

[21] *Florian Hess*. Pairing Lattices [Електронний ресурс] / Florian Hess // Cryptology ePrint Archive: Report 2008/125. – 2008. – Режим доступу: http://eprint.iacr.org/.

[22] *Daniele Micciancio, Oded Regev*. Lattice-based Cryptography // Daniele Micciancio, Oded Regev /In D.J. Bernstein; J. Buchmann; E. Dahmen edit. – «Post Quantum Cryptography». – Springer. – 2009. – P.147-191.

Надійшла до редколегії 2.03.2012

Горбенко Іван Дмитрович, фото та відомості про автора див. на с. 190.

Макутоніна Лідія Вікторівна аспірант кафедри БІТ ХНУРЕ. Область наукових інтересів: асиметричні системи шифрування, криптографічні системи та протоколи, що засновані на ідентифікаторах та алгебраїчних решітках.

УДК 004.056.55

Анализ криптографических алгоритмов на идентификаторах, использующих алгебраические решетки / И.Д. Горбенко, Л.В. Макутонина, // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 200–209.

Приводятся результаты анализа существующих алгоритмов на идентификаторах, использующих алгебраические решетки. Обосновываются выбор параметров, для обеспечения необходимого уровня защищенности. Излагается доказательство безопасности относительно задачи обучения с ошибками в модели случайного оракула.

Ключевые слова: криптографические системы на идентификаторах, алгебраические решетки, алгоритм зашифрования, алгоритм розшифрования, идентификатор пользователя, прообраз выборки, функция с секретом.

Библиогр.: 22 назв.

UDK 004.056.55

Analysis of identity-based cryptographic algorithms using algebraic lattices / I.D. Gorbenko, L.V. Makutonina, // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 200–209.

The results of the analysis of existing identity-based cryptographic algorithms using algebraic lattices are given. The choice of parameters to ensure the necessary level of security is grounded. The proof of safety with respect to the problem of learning with errors in a random oracle model is presented.

Keywords: identity-based encryption systems, algebraic lattices, encryption algorithm, decryption algorithm, user ID, sampling preimage, trapdoor function.

Ref.: 22 items.

УДК 681.3.06

АНАЛИЗ КРИПТОГРАФИЧЕСКИХ СИСТЕМ В ГРУППАХ КОС

Д.А. ПАРШИНА, И.А. МИТЯЕВА, И.Д. ГОРБЕНКО

На протяжении нескольких последних лет заметно вырос интерес к криптографическим приложениям, основанным на преобразованиях в некоммутативных группах. Группы кос в частности представляют особый интерес в силу своей эффективности при обеспечении трудоёмких вычислительных процессов. Различными группами исследователей были предложены протоколы с преобразованиями в группе кос. Данная работа посвящена описанию основных криптографических преобразований в кос-группах, обзору некоторых протоколов, использующих данные преобразования, а также рассмотрению самых распространённых вопросов в этой области.

Ключевые слова: преобразования в группах КОС, механизмы обмена ключами, схемы шифрования, механизмы аутентификации, электронная цифровая подпись, задача поиска сопряжений.

ВВЕДЕНИЕ

В последние годы проявляется интерес к криптографическим преобразованиях в некоммутативных группах КОС[1]. Их особенностью является эффективность при обеспечении трудоёмких вычислительных процессов. Рядом исследователей предложены криптографические протоколы, которые базируются на преобразованиях в группе КОС. В тоже время возможности практического применение преобразований в группах КОС ограничены из-за недостаточного их анализа, как раз в криптографических приложениях. Целью настоящей статьи является рассмотрение и первичный анализ основных криптографических преобразований и криптографических протоколов в КОС-группах, а также рассмотрению проблемных вопросов[1].

1. ОСНОВНЫЕ ПОНЯТИЯ О ГРУППАХ КОС

Коса из n-ломаных нитей – объект который состоит из двух параллельных плоскостей P_0 и P_1 в трёх мерном пространстве R^3, который состоит из упорядоченного множества точек $a_1, a_2, ..., a_n \in P_0$, $b_1, b_2, ..., b_n \in P_1$, и из n – простых ломаных l_1, l_2, ... l_n, которые не пересекаются между собой, пересекая каждую плоскость P_t между P_0 и P_1 и соединяют точки $\{a_i\}$ с точками $\{b_i\}$.

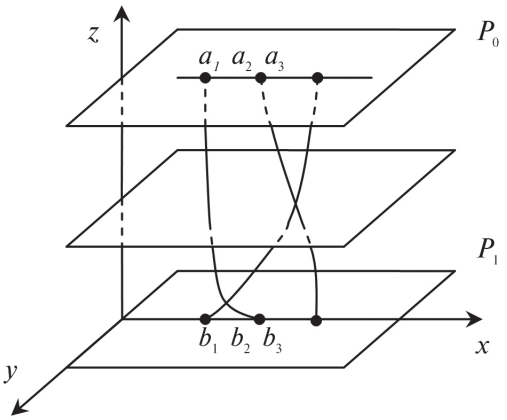


Рис. 1. Графическое представление косы

Косы из n-нитей, аналогично перестановкам владеют природной структурой групп. Пусть есть две косы A и B. Операция умножения КОС определяется как: вертикальное сжимание и расположение одна над одной (рис. 2*а*). Нейтральным элементом в группе КОС является коса с вертикально расположенными нитями (рис. 2*б*). Обратный элемент в группе КОС задаётся вертикальным отображением (рис. 2*в*).

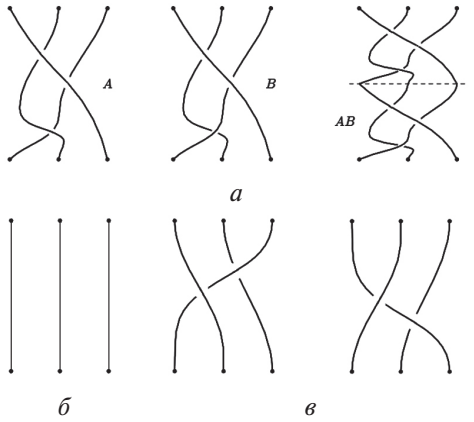


Рис. 2. Операции в группе кос:
а – умножение; *б* – нейтральный элемент;
в – обратный элемент

Фундаментальная коса – $\Delta_n \in B_n$, это коса, алгебраическое представление которой имеет вид $\Delta_n = (\sigma_1 ... \sigma_{n-1})(\sigma_1 ... \sigma_{n-2}) ... \sigma_1$, где σ_i – образующий элемент.

Основные соотношения в группе кос направлены на изменение формы записи, при этом не изменяя изоморфного класса косы.

Дальняя коммутативность – если существует два пересечения, которые находятся на большом расстоянии друг от друга по горизонтали, но близко по вертикали (не существует ни одного пересечения, которое находится выше одного из них, но ниже другого), порядок существующих элементов σ_i и σ_j изменится на σ_j и σ_i:

$$\sigma_i \sigma_j = \sigma_j \sigma_i, \text{ при условии } |i-j| \geq 2. \qquad (1)$$

Второе движение Рейдемейстра – пусть две нити косы находятся на близком расстоянии друг от друга и не пересекаются, тогда одну из этих нитей можно «накласть» на другую, то есть провести сверху другой, что можно описать соотношением:

$$\sigma_i^{-1}\sigma_i = \sigma_i\sigma_i^{-1} = e, \quad (2)$$

где e – нейтральный элемент.

Третье движение Рейдемейстра – движение, которое в теории узлов описывается формулой:

$$\sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1}, \quad (3)$$

при условии $1 \le i \le n-2$.

Если для некоторой косы существуют три точки попарных пересечений трёх разных нитей косы, которые находятся рядом, при этом одна из нитей проходит выше (ниже) других двух, то её можно протянуть над (под) двумя другими (рис. 3).

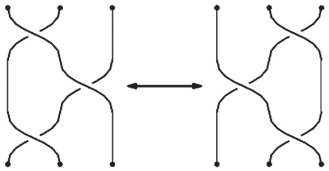


Рис. 3. Третье движение Рейдемейстра

Фундаментальной в теории кос является теорема Артина: группа кос B_n, изоморфна абстрактной группе, порождённой образующими b_1, b_2, ... b_{n-1}, которые удовлетворяют соотношениям (1) – (3). В алгебраическом виде это можно записать так:

$$\left\langle \sigma_1,\ldots,\sigma_{n-1} \left| \begin{array}{c} \sigma_i\sigma_j = \sigma_j\sigma_i \text{ для } |i-j| \ge 2 \\ \sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1} \text{для } |i-j| = 1 \end{array} \right. \right\rangle. \quad (4)$$

2. КРИПТОГРАФИЧЕСКИЕ ПРЕОБРАЗОВАНИЯ В ГРУППАХ КОС

К основным криптографическим преобразования в группах КОС относятся: механизмы обмена ключами, системы шифрования, аутентификации и цифровой подписи[2].

2.1. Механизмы обмена ключами

Среди *механизмов обмена ключами* можно выделить два основных – это протокол Аншеля-Аншеля-Гольдфельда и протокол, аналогичный алгоритму Диффи-Хеллмана. В протоколе Аншеля-Аншеля-Гольдфельда в качестве открытого ключа принимается два набора кос $\{\{p_1, ..., p_n\}, \{q_1, ..., q_m\}\} \in B_n$. Секретный ключ U, принадлежащий A, состоит из l нитей и их инверсий. Аналогично секретный ключ V, принадлежащий B, состоит из m нитей и их инверсий. Обмен происходит следующим образом :

1. А генерирует косу $s = u(p_1, ..., p_l)$, и использует её, чтобы сгенерировать сопряжённые $q_1' = sq_1s^{-1}, ..., q_m' = sq_ms^{-1}$; пересылает $q_1'...q_m'$.

2. В генерирует косу $r = v(q_1, \ldots, q_m)$, и использует её, чтобы сгенерировать сопряжённые $p_1' = rp_1r^{-1}, ..., p_l' = rp_lr^{-1}$; пересылает $p_1'...p_l'$.

3. А вычисляет $t_A = su(p_1, ..., p_l)^{-1}$.

4. В вычисляет $t_B = v(q_1, ..., q_m)r^{-1}$.

Искомый ключ $t_A = t_B$[3].

Протокол, который предложен К.Н. Ко, базируется на классическом протоколе Диффи — Хеллмана. Здесь, открытый ключ p это определённая коса в группе B_n. Секретный ключ принадлежащий A представляет собой косу s из подгруппы LB_n, а секретный ключ B – косу r из подгруппы UB_n. Обмен ключами происходит таким образом:

1. А генерирует сопряжение $p' = sps^{-1}$ и пересылает его B;

2. В генерирует сопряжение $p'' = rpr^{-1}$ и пересылает его A;

3. А вычисляет $t_A = sp''s^{-1}$;

4. В вычисляет $t_B = rp'r^{-1}$;

Искомый ключ $t_A = t_B$.

2.2. Схема (метод) шифрования

Данная схема была предложена К.Н. Ко. Пусть есть группа кос B_n, и её подгруппа LB_n (соответственно UB_n), порождённая элементами σ_1, ..., σ_{m-1} (соответственно σ_{m+1}, ..., σ_{n-1}) из $m = n/2$. Каждая коса из LB_n будет коммутативна каждой косе из UB_n. h – безколизийная однонаправленная хеш-функция.

$$(h(b_1) \ne h(b_2)),\ B_n \to \{0, 1\}^N.$$

Алгоритм генерации ключевой пары:

1. Выбирается открытая коса $p \in B_n$;
2. Выбирается персональный ключ $s \in LB_n$;
3. Вычисляется открытый ключ $p' = sps^{-1}$;
4. В качестве персонального ключа используется s, в качестве открытого ключа пара (p, p')

Алгоритм зашифрования:

Вход: открытый ключ (p, p'), сообщение m из пространства $\{0, 1\}^N$, h – хеш-функция.

Выход: криптограмма e.

1. Абонент выбирает случайную косу r из $U B_n$, и вычисляет $p'' = rpr^{-1}$

2. Зашифровывает сообщение: $e = m \oplus h(rp'r^{-1})$

3. В качестве криптограммы на выход подаётся (e, p'').

Алгоритм расшифрования:

Вход: персональный ключ s, криптограмма (e, p''), h – хеш-функция.

Выход: сообщение m.

Абонент используя персональный ключ s вычисляет $m = e \oplus h(sp''s^{-1})$[4].

2.3. Механизмы аутентификации

Как и в предыдущих системах, открытый ключ – это пара сопряжённых кос (p, p'), причём $p' = sps^{-1}$, принадлежащих группе B_n, сопряжённая коса s является секретным ключом A. В отличие от предыдущих систем и p и s принадлежат группе B_n, т.е мы не можем предположить, что s принадлежит какой-нибудь из подгрупп LB_n или же UB_n. Однако по прежнему предположим, что h-это односторонняя хэш-функция, в которой не происходит коллизий, заданная в группе B_n как $\{0,1\}^N$. Процедура аутентификации заключается в повторении k раз следующих трёх шагов:

1. А выбирает случайную косу r, принадлежащую B_n и пересылает запрос $x = h(rp'r^{-1})$;

2. В выбирает случайный бит c и пересылает его A;

3. Для $c = 0$, А пересылает $y = r$, и B проверяет $x = h(yp'y^{-1})$;

4. Для c=1, A пересылает $y=rs$, и B проверяет $x= h(yp'y^{-1})$.

2.4. Электронная цифровая подпись

Две системы электронной подписи были предложны K.H.Ko: применение второй схемы рекомендовано автором, однако на примере первой легче разобраться в самом алгоритме подписи, он является более наглядным и легко читаемым. Как и ранее открытый ключ представляет собой пару кос (p, p'), $p' = sps^{-1}$, принадлежащих группе B_n, а сопряжённая им коса s, принадлежащая B_n, является персональным ключом A. Будем использовать однонаправленную хэш-функцию H из $\{0,1\}^*$ в B_n. На первом шаге выполняются следующие действия:

1. A подписывает сообщение m при помощи $q_1' = sq_1s^{-1}$, где $q=H(m)$;

2. В проверяет $q' \approx q, p'q' \approx pq$.

Если A использует секретный ключ s, то получаем $q_1' = sq_1s^{-1}$, и $p'q' = spqs^{-1}$, то есть подпись принята. Возможная слабость данной системы может быть обусловлена тем, что возможно возникающие повторения могут раскрыть достаточно большое кол-во сопряжённых пар $\left(q_i, q_i'\right)$, связанных с начальным сопряжением s, что делает возможным осуществление атаки на такую систему. Чтобы избежать этого, автор впоследствии несколько изменил общую схему путём включения дополнительных случайных кос.

3. ОЦЕНКА КРИПТОГРАФИЧЕСКОЙ СТОЙКОСТИ

Описанные выше преобразования в немалой степени завися от решения следующих задач.

3.1. Задача поиска корня

Пусть $(x,y) \in B_n \times B_n$ такие, что $y = x^c$, $c \in N$, $c \geq 2$, N – множество натуральных чисел. Задача нахождения корня состоит в нахождении такой косы $b \in B_n$, чтобы $y = b^c$, $c \geq 2$.

3.2. Задача декомпозиции кос

Пусть $(x,y) \in B_n \times B_n$ и $y = a_1xa_2$ для некоторых $(a_1, a_2) \in B_n \times B_n$. Задача декомпозиции кос состоит в нахождении такой пары $(b_1, b_2) \in B_n \times B_n$, чтобы $y = b_1xb_2$.

3.3. Криптографическое допущение

В данной схеме мы рассматриваем группу кос B_n, порожденную $\{\sigma_1,\sigma_2,...,\sigma_n\}$ и ее подгруппами

$$LB_n = \{\sigma_1,\sigma_2,...,\sigma_{n/2-1}\}$$

$$RB_n = \{\sigma_{n/2+1},\sigma_{n/2+2},...,\sigma_{n-1}\}$$

Связь этих групп определяется как:

$$\sigma_i\sigma_j = \sigma_j\sigma_i, \ |i-j| > 1$$

$$\sigma_i\sigma_j\sigma_i = \sigma_j\sigma_i\sigma_j, \ |i-j| = 1$$

Т.е. мы имеем коммутативное свойство $\alpha\beta = \beta\alpha$ для любых $\alpha \in LB_n$ и $\beta \in RB_n$.

Пусть H_1: $B_n \rightarrow \{0,1\}^k$ – идеальная функция нахождения $\{0,1\}^k$ из косы.

Пусть H_2: $\{0,1\}^k \rightarrow B_n$ – идеальная функция нахождения косы из $\{0,1\}^k$.

Пусть c: $\{0,1\}^k \rightarrow B_n$.

Стойкость криптосистем с использованием кос-групп основывается на следующих проблемах:

1. Задача поиска сопряжений (CSP):

Пусть $(x, y) \in B_n \times B_n$ такие, что $y = a^{-1}xa$, где $a \in B_n$ или одной из подгрупп B_n. Задача – найти такое b, что $y = b^{-1}xb$.

2. Задача одновременного поиска множества сопряжений (MSCSP):

Пусть $(x_1,a^{-1}x_1a)...(x_r,a^{-1}x_ra) \in B_n \times B_n$ такие, что $y = a^{-1}xa$, где $a \in B_n$ или одной из подгрупп B_n. Задача – найти такое b, что $y = b^{-1}x_1b = a^{-1}x_1a$, ..., $b^{-1}x_rb = a^{-1}x_ra$.

3. Задача декомпозиции (*BDP*):

Пусть $(x, y) \in B_n \times B_n$ такие, что $y = a_1xa_2$ для $(a_1,a_2) \in LB_n \times LB_n$. Задача – найти пару $(b_1,b_2) \in LB_n \times LB_n$ такую, что $y = b_1xb_2$.

4. Задача одновременной множественной декомпозиции (MSBDP):

Пусть $(x_1,a_1x_1a_2)$ ... $(x_r,a_1x_ra_2) \in B_n \times B_n$ для $(a_1,a_2) \in LB_n \times LB_n$. Задача – найти пару $(b_1,b_2) \in LB_n \times LB_n$ такую, что $y = b_1x_1b_2 = a_1x_1a_2$, ..., $b_1x_rb_2 = a_1x_ra_2$.

5. Задача поиска корня (RP):

Пусть x = a^p, где a, $x \in B_n$ и $p \in N$. Задача поиска для экспоненты p – найти такую косу $b \in B_n$, чтобы $b^p = x$.

6. Задача выбора сопряженных элементов (*CDP*):

Пусть $(x, y) \in B_n \times B_n$. Задача – установить, являются ли x и y сопряженными, т.е. установить, существует ли такое $a \in B_n$ или одной из подгрупп B_n, что $y = a^{-1}xa$.

Исходя из вышеприведенного, рассмотрим три основные разновидности атак на криптосистемы, основанные на преобразованиях в группах кос:

1) использование решения задачи поиска сопряжений;

2) использование вероятностного подхода в B_n;

3) использование вспомогательной группы, как правило, в представлении Бурау[1].

3.4. Решение задачи поиска сопряжений

Наиболее очевидный способ атаки на кос-криптосистемы – решение задачи поиска сопряжений в B_n, который стал известен благодаря основополагающей работе Гарсайда. Последующие уточнения метода значительно улучшили его алгоритмическую эффективность.

Метод Гарсайда для решения задачи поиска сопряжений в B_n состоит в привязке к каждой косе b характерного конечного набора сопряжений b, называемого высшим множеством. Эль-Рифай и Мортон предложили заменить высшее множество его подмножеством – супер высшим множеством(*SSS*). Супер высшее множество меньше, следовательно, его легче определить. Под *SSS* подразумевается множество всех сопряжений b минимально возможной запутанности.

Для каждой косы b супер высшее множество конечно и алгоритмически вычислимо.

Две косы b и b' сопряжены тогда и только тогда, когда их *SSS*. Таким образом, предполагаем разрешимость задачи поиска сопряжений в B_n. В действительности, известны и более точные результаты. Введем следующее определение: фундаментальная коса – $\Delta_n \in B_n$, это коса, алгебраическая запись которой имеет вид:

$$\Delta_n = (\sigma_1 ... \sigma_{n-1})\,(\sigma_1 ... \sigma_{n-2}) ... \sigma_1.$$

Геометрический пример приведен для косы Δ_4, где любые две нити пересекаются положительно, кроме одной (рис. 1).

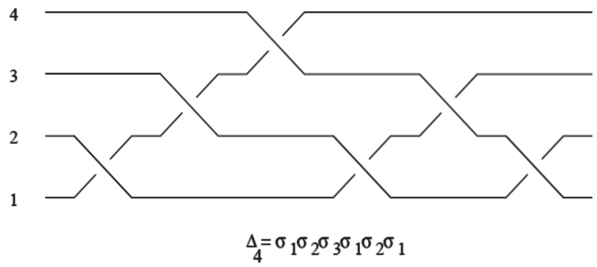


Рис. 4. Фундаментальная коса для Δ_4

Предположим, что b – коса в B_n и $(k; b_1 ..., b_r)$ – её нормальная форма. Если косы $\partial_+(b)$ и $\partial_-(b)$ определяются как

$$\partial_+(b) = \Delta_n^k b_2 ... b_r \varphi_n^k(b_1),$$

$$\partial_-(b) = \Delta_n^k \varphi_n^k(b_r) b_1 ... b_{r-1}, \qquad (5)$$

где φ_n – флип-автоморфизм, отображающий σ_i в σ_{n-i} для каждого i; считается что $\partial_+(b)$ (соответственно $\partial_-(b)$) получена циклированием (дециклированием) из b.

Косы $\partial_+(b)$ и $\partial_-(b)$ – сопряжения b. Дело в том, что если b – коса в B_n, не принадлежащая супер высшему множеству b, т.е. не имеет минимальной запутанности в этом классе сопряжений, тогда циклированием или дециклированием максимум $n(n-1)/2$ раз можно найти сопряжение b точно меньшей запутанности. Таким образом, повторяя эти действия, после конечного числа шагов мы получим сопряжение b^* для b, лежащее в супер высшем множестве b.

Приведем полную процедуру принятия решения о сопряженности кос b и b', проиллюстрированную на рис. 5:

1) Используя циклирование (cycling) и дециклирование (decycling), найти b^* для b, лежащую в супер высшем множестве (*SSS*) b;

2) Используя циклирование и дециклирование, найти b^* для b', лежащую в $SSS(b')$;

3) Определить $SSS(b)$, насыщая $\{b^*\}$ простыми сопряжениями;

4) b и b' будут сопряженными, если b'^* принадлежит

Отслеживая сопряжение кос на каждом шагу, можно не только определить, являются ли b и b' сопряженными, но также получить сопряжение, если оно существует, т.е. если b и b' сопряжены. Таким образом, решаются две задачи: задача сопряжения и задача поиска сопряжений в B_n[2].

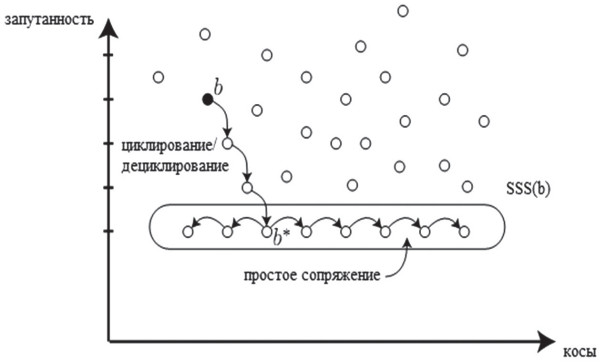


Рис. 5. Решение задачи сопряжения: определение SSS и его перечисление (точки показывают сопряжения b)

Что касается сложности, так как циклирование и дециклирование постоянное количество раз гарантирует, что нормальная длина будет уменьшаться, если это возможно, нахождение сопряжения в *SSS* имеет линейную сложность по сравнению со сложностью для исходной косы. Потом остается только сложность перечисления $SSS(b)$.

Совсем недавно В. Гебхардт предложил новое совершенствование. Это совершенствование состоит в замене *SSS* еще меньшим множеством, называемым ультра высшим множеством (*USS*). Рассмотрим действие циклирования на *USS*: начиная с косы b в ее *SSS*, не обязательно возвращаться к исходной b в циклировании *SSS*, но, безусловно, циклирование, в конечном счете, становится периодичным. Таким образом, можно разделить *SSS* на несколько орбит, состоящих из цикличных частей и остатков (рис. 6).

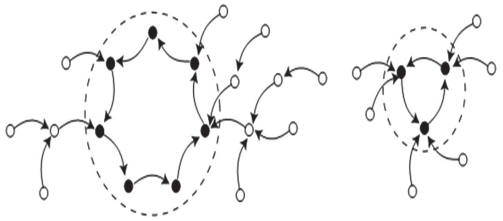


Рис. 6. Действие циклирования в *SSS*; черным показаны элементы *USS*

Гебхардт определяет ультра высшее множество как объединение цикличных частей орбиты. По определению *USS* является подмножеством *SSS*, и Гебхардт показывает, что *USS* может быть использовано вместо *SSS*: как и для *SSS*, элементы *USS* легко определить, а потом подсчитать и все *USS*, используя минимальные простые элементы. Дело в том, что размер *USS* обычно гораздо меньше *SSS*, типично его размер линеен относительно длины исходной косы, тогда как размер *SSS* экспоненциален. В таких случаях *USS* можно определить быстро и проблема сопряжения будет решена. На данный момент это не доказано, но сложность метода может быть сведена к полиномиальной.

3.5. Атаки, основанные на длине

Помимо использования конкретного решения задачи поиска сопряжений, также кос-криптосистемы можно атаковать, используя вероятностный эвристический подход: всякий раз, когда вероятность успеха более чем незначительна, этого может быть достаточно для того, чтоб поставить под угрозу кос-криптосистему. Основанные на длине атаки относятся к этому семейству. Общий принцип таких атак состоит в попытке получить сопряжение для пары (p, p'), начиная с p', которая должна быть получена из p и многократно сопрягающаяся с p' в новую косу $tp't^{-1}$ так, что длина или запутанность $tp't^{-1}$ будет минимальной.

При осуществлении атаки проверяется, случается ли, что новое сопряжение $tp't^{-1}$ равно p. Атака особо применима к протоколам обмена ключами, основанным на задаче одновременного поиска множества сопряжений, потому что, в данном случае, злоумышленник знает несколько пар сопряженных кос, связанных с одной и той же сопряженной косой. Атака, описанная Хофхайнцем и Штайнвандтом, аналогична, но она включает в себя еще один шаг, и поэтому является более мощной. Вместо проверки, является ли $tp't^{-1}$ равным p, злоумышленник проверяет, чтобы «расстояние перестановки» между $tp't^{-1}$ и p не превышало 1, т.е. пытается найти такую перестановку f, что $tp't^{-1}$ равно простому сопряжению fpf^{-1}. Нахождение возможных перестановок является очень легким, так как оно сводится к решению задачи поиска сопряжений в симметричной группе S_n. При этом улучшении вероятность успешного осуществления атаки достигает 99% для протокола согласования ключей Аншеля-Аншеля-Гольдфельда в B_{80} при $l = m = 20$ и исходными косами p_i и q_j длины 5 или 10[3].

3.6. Атаки, основанные на линейных представлениях

Третий способ атаки кос-криптосистем – использование линейного представления кос-групп, т.е. отображение кос-групп в группы матриц. Так как задача сопряжения в линейной группе легка, так что можно думать о решении задачи сопряженности таким способом.

Наиболее известным представлением кос-групп B_n является представление Бурау, линейное представление со значениями $GL_n(Z[t, t^{-1}])$. Представление Бурау для B_n, как известно, неточно для $n \geq 5$, но ядро очень мало, потому вероятность того что различные косы примут один и тот же образ Бурау незначительна [3].

Рассмотрим также результаты, полученные в результате анализа безопасности подобных систем.

Только уполномоченная сторона может проверить подпись (R_1, R_2, S_1, S_2, S_3, δ): *Только уполномоченная сторона владеет секретным ключом b, используя который вычисляет:*

$$R_3 = bR_1b^{-1}, m = H_1(R_3) \oplus S_3$$

$$R_4 = bR_2b^{-1}, S_4 = H_2\left[H_1(R_4) \oplus m\right]$$

$$\theta = b^{-1}S_1b.$$

Затем проверяет равенство $\delta = \left[S_4\theta\right]^c$. Если неуполномоченная сторона хочет проверить (R_1, R_2, S_1, S_2, S_3, δ), что она должна вычислить R_3, R_4, θ, что невозможно сделать без секретного ключа b.

Любой злоумышленник не может атаковать подпись

Любой злоумышленник не может получить секретный ключ (u_i, v_i), без знания которого невозможно вычислить R_1, R_2, R_3, R_4, S_1, S_2, поскольку вычисления основываются на задачах поиска сопряжений и декомпозиции кос. Предположим, злоумышленник подделал $\delta = \left[S_4d\right]^c$, но он не может определить S_4d из-за сложности задачи поиска корня. Таким образом, любой противник не может ни вычислить сообщение, ни выполнять проверки.

Никто не может снять подпись за исключением авторизованной группы T.

Не зная секретные ключи u_i, a, никто не может вычислить сертификат авторизации y_i и проверить равенство $S_4^c = \delta\theta^{'-1}$. Авторизованная группа T, имея секретный ключ a, может вычислить сертификат авторизации y_i и θ' и проверить равентво $S_4^c = \delta\theta^{'-1}$. Если равенство имеет место, авторизованная группа объявляет, что подпись осуществляется P_i (i=1,2,3,...,k).

Анализ рассмотренных криптографических систем показывает, что разработка алгоритмов, использующих группы кос является перспективным направлением в развитии современной криптографии [4]. Основные характеристики подобных систем приведены в табл. 1.

Таблица1

Основные характеристики криптографических систем, базирующихся на группах кос

Входящее сообщение, бит	$pn\log(n)$
Зашифрованное сообщение, бит	$4pn\log(n)$
Скорость зашифрования, операций	$O(p^{2n}\log(n))$
Скорость расшифрования, операций	$O(p^{2n}\log(n))$
Длина персонального ключа, бит	$0{,}5pn\log(n)$
Длина открытого ключа, бит	$3pn\log(n)$
Сложность атаки «грубая сила»	$((n/2)!)^p = \exp(0{,}5pn\log(n))$

ВЫВОДЫ

В целом важным фактором, влияющим на возможности осуществления атаки является способ генерации ключей. Так, например, атака Гебхардта возможна лишь при достаточно малом USS, что не всегда соответствует действительности. Из вышеизложенного следует, что вычисление $p' = sps^{-1}$ с исходной косой p не является лучшим способом генерации пары сопряженных кос. Действительно, установление ряда ограничений на ключи – довольно распространенная

ситуация, существует всего несколько криптосистем, в которых ключи могут быть выбраны в случайном порядке. Поэтому даже если некоторые авторы утверждают, что существующие атаки полностью нивелируют криптографию в группах кос, на данный момент, более разумным кажется заключить, что необходимо приложить больше усилий для построения доказуемо стойких крипо алгоритмов или же предоставлении доказательств того, что построение подобных крипто алгоритмов невозможно.

Литература

[1] *D. Garber, S. Kaplan, M. Teicher, B. Tsaban and U. Vishne*, Length-based conjugacy search in the braid group, Contemp. Math. 418 (2006), 75–87.

[2] *E. Artin,* Theory of Braids, Ann. of Math. 48 (1947) 101–126.

[3] *I. Anshel, M. Anshel, & D. Goldfeld,* An algebraic method for public-key cryptography, Math.Research Letters 6 (1999) 287–291.

[4] *J.C. Cha, K.H. Ko, S.J. Lee, J.W. Han, J.H. Cheon,* An efficient implementation of braid groups, AsiaCrypt 2001, Springer Lect. Notes in Comput. Sci., 2048 (2001) 144–156.

Поступила в редколлегию 14.03.2012

Паршина Дарья Андреевна, студентка кафедры БИТ. Область научных интересов: криптографические протоколы в группах кос.

Митяева Ирина Андреевна, студентка кафедры БИТ. Область научных интересов: анализ криптографических протоколов в группах кос.

Горбенко Иван Дмитриевич, фото и сведения об авторе см. на с. 190.

УДК 681.3.06

Аналіз криптографічних систем в групах КОС/ Д.А. Паршина, І.А. Мітяєва, І.Д. Горбенко // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 210–215.

Протягом кількох останніх років помітно зростає інтерес до криптографічних додатків, заснованих на перетвореннях у некомутативних групах. Групи КОС зокрема представляють особливий інтерес завдяки своїй ефективності при забезпеченні трудомістких обчислювальних процесів. Різними групами дослідників були запропоновані протоколи з перетвореннями в групах КОС. Дана робота присвячена опису основних криптографічних перетворень в кос-групах, огляду деяких протоколів, що використовують дані перетворення, а також розгляду найпоширеніших питань у цій галузі.

Ключові слова: перетворення в групах КОС, механізми обміну ключами, схеми шифрування, механізми автентифікації, електронний цифровий підпис, завдання пошуку сполучень

Табл. 1. Іл. 6. Бібліогр.: 4 найм.

UDC 681.3.06

Analysis of cryptographic system in braid groups / D.A. Parshina, I.A. Mityaeva, I.D. Gorbenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 210–215.

The past several years have seen an explosion of interest in cryptographic applications based on transformations in non-communicative groups. Braid groups are of particular interest due to their efficiency in providing labour-consuming computational processes. Different groups of researchers have proposed protocols with transformations in the braid groups. The paper is devoted to describing the main cryptographic transformations in the braid groups, reviewing some protocols using the given transformations as well as considering the most common questions in this field.

Keywords: transformations in braid groups, key exchange mechanisms, encryption schemes, authentication mechanisms, digital signature, conjugation search problem.

Tab. 1. Fig. 6. Ref.: 4 items.

УДК 621. 391:519.2:519.7

АТАКА СПЕЦІАЛЬНОГО ВИДУ НА NTRU

М.Ф. БОНДАРЕНКО, Д.С. БАЛАГУРА, Д.В. ІВАНЕНКО

Наводиться порядок шифрування та розшифрування алгоритму NTRU. Розглядається атака спеціального виду за часом, демонструється можливість отримати знання про секретний ключ при відомій кількості звернень до геш-функції.

Ключові слова: алгоритм NTRU, атака спеціального виду, шифрування, розшифрування, шифр-текст, геш-функція, поліном.

ВСТУП

Алгоритм NTRUEncrypt(NTRU) був розроблений ще у середині 90-х математиками Jeffrey-Hoffstein, JillPipher и Joseph H. Silverman [1]. Він був заснований на решітчастій криптосистемі, яка у майбутньому могла би протидіяти атакам з використанням квантових комп'ютерів. Але у запропонованому на той час алгоритмі були суттєві недоліки: NTRU програвав у швидкодії та у продуктивності RSA [2] та криптосистемам на еліптичних кривих [2], тому одразу не отримав широкого розповсюдження. На сьогоднішній день основні недоліки, за твердженнями спеціалістів RSALabs та за результатами незалежних досліджень усунені. Фактом визнання якісних показників є прийняття NTRU технологічним стандартом для фінансових транзакцій та визнання Accredited Standards Committee X9 NTRU найшвидшим алгоритм асиметричного шифрування.

Зважаючи на швидке розповсюдження та визнання NTRU постає питання глибшого вивчення аспектів стійкості протоколів до будь-яких атак. Метою статті є аналіз можливості застосування відносно NTRU деякої атаки спеціального виду, а також обґрунтування пропозицій щодо попередження атаки спеціального виду за часом

1. ЗАГАЛЬНА ХАРАКТЕРИСТИКА АЛГОРИТМУ NTRU

1.1. Зашифрування

У зашифруванні використовується дві геш-функції. Позначимо їх G та H, на практиці зазвичай використовується SHA-1 або SHA-256. Зашифрування здійснюється за таким алгоритмом:

Крок 1: Візьмемо відкритий текст M та переведемо його у β_N(бінарний поліном):

$$M \in \beta_N,$$

$$\beta_N := \{d_1 X^{N-1} + d_2 X^{N-2} + \ldots + d_{N-1} X^1 + d_N 1 \in R, d_i = \{0,1\}\}$$

$$\beta_N(d) := \{d_1 X^{N-1} + \ldots + d_N 1 \in R, d_i = \{0,1\} \text{ та } \sum d_i = d\} \quad (8)$$

Крок 2: за допомогою геш-функції G рандомізуємо текст M додаючи до нього бінарний поліном (d_r):

$$r = G(M) \in \beta_N(d_r). \quad (1)$$

Крок 3: на цьому етапі потрібно розрахувати замасковане повідомлення m':

$$m' = M \oplus H(r \cdot h \bmod q). \quad (2)$$

Крок 4: на останньому етапі отримуємо зашифрований текст e:

$$e = (r \cdot h + m') \bmod q. \quad (3)$$

1.2. Розшифрування

На етапі розшифрування відбувається не тільки розшифрування, але й перевірка справжності отриманої пари (m', e). Алгоритм розшифрування також використовує дві геш-функції G та H.

Крок 1: Алгоритм розшифрування відновлює отримане m':

$$m' = ((f \cdot e \bmod q) \bmod p) \cdot f_p^{-1} \bmod p. \quad (4)$$

Крок 2: На цьому етапі відновлюється текст M:

$$M = m' \oplus H(e - m' \bmod q). \quad (5)$$

Крок 3:Після визначення M та m', необхідно відтворити r:

$$r = G(M). \quad (6)$$

Крок 4: на останньому етапі перевіряємо чи (m', e) відповідає парі NTRU:

$$e = r \cdot h + m' \bmod q. \quad (7)$$

2. АТАКА ЗА ЧАСОМ НА NTRU

Нижче розглядається атака спеціального виду на особистий ключ алгоритму NTRU - атака за часом, яка базується на підрахуванні кількості звернень до геш-функції. Використовуючи слабкі сторони структури алгоритму, застосовуючи вже визначені спеціальні канали, які отримали назву атаки спеціального виду, можливо найти таємну інформацію.

2.1. Атака за часом на шифр-текст

При зашифруванні використовують сеансовий ключ r, який додавався до нашого повідомлення M. Для відтворення r потрібно підрахувати кількість звернень до геш-функції пари (m', e) відповідного повідомлення. Кількість цих звернень може бути різною для різних повідомлень. Тобто кількість звернень до геш-функції визначається для кожного повідомлення окремо. Зловмисник повинен виміряти кількість звернень до геш-функції підчас розшифрування шифр-тексту e. Існує таке число K, що є кількістю звернень до геш-функції, необхідних для визначення r, яке може дорівнюватися або k або k+1. На

наступному етапі потрібно визначити вихідне $r(m', e)$ для відповідної пари (m', e) кожного повідомлення з алгоритму розшифрування:

$$r(m',e) = G((m' + H(e - m' \bmod q)) \bmod 2. \quad (8)$$

Визначимо, що $\beta_N(m',e) \in \{0,1\}$. *Нуль* встановлюється у випадку коли потрібно більше ніж K геш значень при створенні $r(m', e)$, та *одиниця*, якщо потрібно K геш значень. За таких умов отримуємо $(X^i m', X^i e)$ при $i = 0,1...$. Зважаючи на вищезазначене, можна дати повне визначення функції атаки за часом. Це бінарний вектор, який має наступний вид:

$$T(m',e) = ((\beta(m',e), \beta(Xm', Xe), \beta(X^2 m', X^2 e),... \\ ..., \beta(X^{N-1} m', X^{N-1} e)). \quad (9)$$

Функція атаки за часом надасть нам кількість звернень до геш-функції потрібних для кожної пари (m', e). Вище було зазначено, що кількість звернень для різних повідомлень різна. Тому можна визначити , з якою імовірністю дві пари матимуть однакову функцію атаки за часом та відповідно і кількість звернень до геш-функцій. Припустимо, що P імовірність того, що для випадкової пари потрібно K звернень. Також припустимо, що $1-P$ імовірність іншої випадкової пари (m_2', e_2), якій потрібно як мінімум K звернень. Якщо P досить велика, то імовірність того, що дві пари матимуть ту ж функцію атаки за часом досить мала. Точніше імовірність може бути представлена наступною формулою:

$$\Pr ob(T(m_1', e_1') = T(m_2', e_2)) = (1 - 2P + 2p^2)^N . \quad (10)$$

Формула (10) може бути представлена у іншому вигляді. Представимо функцію атаки за часом у вигляді бінарного вектору довжиною N. Нехай P імовірність того що в i-й позиції бінарного вектору ми отримаємо 0. Ми отримаємо імовірність $1-P$ того що в i-й позиції буде 1. Тоді імовірність того, що в першій позиції двох випадкових функцій атак за часом буде таж сама:

$$\Pr ob(both = 0) + \Pr ob(both + 1) = p^2 + (1 - P)^2 . \quad (11)$$

Для того, щоб дві функції атаки за часом були ідентичні, вони повинні мати в усіх N позиціях деяке число (0 або 1). Таким чином, ми маємо:

$$\Pr ob(T(m_1', e_1') = T(m_2', e_2)) = (1 - 2P + 2p^2)^N . \quad (12)$$

Функція атаки за часом – це представлення ключа NTRU, за допомогою атаки за часом. Далі розглянемо цю атаку більш детально.

2.2. Атака за часом, яка базується на різній кількості звернень до геш-функції

Розглянемо ситуацію, коли є дві сторони *А* та *Б*, вони між собою обмінюються інформацією, припустимо, що *Б* – зловмисник, який намагається отримати секретний ключ іншої сторони. Щоб виконати своє завдання він може використовувати атаку спеціального виду за часом. Зловмисник вибирає множину ε (шифр-текст), яка є набором поліномів за *modq*. Далі *Б* повинен вибрати таке повідомлення M, щоб воно складалося з множини повідомлень створених A, та мало наступний вигляд:

$$\{((f \cdot e \bmod q) \bmod 2 \cdot (f^{-1} \bmod 2) : e \in \varepsilon\} .$$

Більш того, припустимо, що імовірність того, що наведене повідомлення створене стороною A буде знайдено у множині M дуже велика. Сторона *Б* проводить атаку та створює таблицю з функцією атаки для кожної пари $M \times \varepsilon$. Іншими словами він створює список-пошук бінарного вектору:

$$(T(m',e)) : m \in M,\ e \in \varepsilon .$$

Атака починається, коли *Б* посилає *А* деяке випадкове $e \in \varepsilon$. Коли шифр-текст буде відправлений, *Б* починає записувати як довго *А* буде розшифровувати цей шифр-текст. Якщо навіть відправлено підроблений шифр-текст, який у випадку успішної перевірки буде відхилено, це не вплине на головну задачу зловмисника, тому що зловмисник зацікавлений у часовій інформації. Тобто інформації, яка пов'язана з можливістю виміряти кількість звернень до геш-функції. Іншими словами він буде знати значення $m'(e)$, яке може бути представлене у вигляді:

$$((f \cdot e \bmod q) \bmod 2) \cdot (f^{-1} \bmod 2) . \quad (13)$$

Стороні *Б* не відомо значення $m'(e)$. Він знає тільки значення $\beta(m'(e), e)$, яке буде використовуватися після того, як буде порівняно зі значеннями отриманими від *А*, попередньо вирахуваними та записаними у таблицю, яка була створена *Б*. Для цього *Б* потрібно $N-1$ записів, які будуть зроблені функцією атаки. Ці значення він отримає після відправлення одного за іншим наступних поліномів:

$$Xe, X^2 e, X^3 e, ..., X^{N-1} e . \quad (14)$$

Після відправки шифр-тексту, зловмисник отримає значення $\beta(m'(X^i e), X^i e)$. Візьмемо випадкове $m'(X^i e)$. Нехай ми знаємо, що чому дорівнює $m'(e)$, тоді можна застосувати це також й до виразу $m'(X^i e)$:

$$m'(X^i e) = \left(\left(f \cdot X^i e \bmod q \right) \bmod 2 \right) \cdot \left(f^{-1} \bmod 2 \right),$$

оскільки усі X дорівнюють 0 чи 1, то ми можемо взяти таке X^i, що отримаємо

$$X^i \cdot ((f \cdot e \bmod q) \bmod 2) \cdot \left(f^{-1} \bmod 2 \right),$$

що насправді буде $X^i m'(e)$.

Сторона Б, знає функцію атаки $T(m'(e), e)$ від $(m'(e), e)$. На наступному етапі повинен відбутись пошук у сформованому списку, у якому з високою імовірністю знаходиться невелика кількість можливих пар із отриманих стороною *Б* пар. Насправді він має щось більше, наприклад *Б* має два полінома e, m' де *А* розшифровує e взявши другий, поліном m'. З цього зловмисник вираховує тільки $m' \cdot f \equiv (f \cdot e \bmod q) \bmod 2$, причому e та m' він знає.

В наступному підрозділі наводиться, як зловмисник може за допомогою наведеної атаки

отримати перевагу щодо визначення особистого(секретного) ключа *A*. Далі, використання секретного ключа *f* буде залежить від значення *e*. Наприклад, якщо елементи ε складаються із поліномів з ненульовими коефіцієнтами, то вираз:

$$m' \cdot f \equiv \left(f \cdot e \bmod q\right) \bmod 2$$

може дати інформацію відносно відстані між ненульовими коефіцієнтами *f*. В наступному підрозділі, описується специфічна множина ε у використанні атаки за часом на практиці, коли ключ *f* має форму $f = 1 + 2F$.

2.3. Атака за часом у випадку $f=1+2f$

З наведеного вище відомо, що у більшості випадків параметр *p* дорівнює 2. Це значить, що ***q*** непарне, і далі секретний ключ $f = 1 + 2F$ для деякого бінарного поліному $F \in \beta_N(d_F)$. Після підрахування, отримуємо, що $F = \sum F_i X^j$, де $F_j \in \{0,1\}$. Визначимо новий параметр λ = [q/4]. В нашому випадку відправною точкою буде, коли Б відправить А шифр-текст *e,* але у такому випадку ми будемо мати

$$\left\{\lambda + \lambda X^i : 1 \le i < N\right\}.$$

Це значить, що випадкове значення *e* теж поліном, але в цьому випадку з двома коефіцієнтами рівними λ та всі іншими коефіцієнтами –0.

Стисло розглянемо, як *Б* проводить атаку на секретний ключ А:

1. Вибираємо змінну δ
2. Нехай

$$\varepsilon = \left\{e_i = \lambda + \lambda X^i : 0 \le i \le (n-1/2)\right\} \text{та}$$

$$M = \beta_N (0 < d \le \delta).$$

Розрахуємо та збережемо усі значення функції атаки $T(m', e)$.

3. Від А передано $X^j e_i$, де j = 0, 1, ·, N – 1. Зробимо деяке число розшифрування, заміряючи це функцією $T(m'(e_i), e_i)$.

4. Далі зробимо пошук можливих кандидатів.

5. Використовуючи отриманні значення $m'(e_i)$, відновлюємо *F*.

В цей момент необхідно знайти можливе значення $m'(e_i)$, що отримане на кроці 2, тобто (1). З цією метою здійснимо розшифрування, як це робить *A*. З початку *A* вираховує:

$$a = f \cdot e \bmod q \equiv (1 + 2F) \cdot \left(\lambda + \lambda X^i\right) (\bmod q) \equiv$$

$$\equiv \lambda + \lambda X^i + \sum_{j=0}^{N-1} 2\lambda \left(F_j + F_{j-i}\right) X^i.$$

Нехай для *j*-го коефіцієнту *a*, ми маємо наступні функції:

$$a_j = \begin{cases} \lambda(1 + 2F_o + F_{-i} \bmod q) \ \textit{if } j = 0, \\ \lambda(1 + 2F_i + 2F_o) \bmod q \ \textit{if } j = i, \\ \lambda(2F_i + 2F_{j-i}) \bmod q \ \textit{if } j \ne 0, i. \end{cases}$$

Розглядати функцію вище ми повинні чітко, так як λ =2[*q*/8], це значення λ не набагато більше ніж *q*/4, и тому права сторона цього виразу приймає значення менше, ніж *q*–1. Тому не потрібно зменшувати $a \,(mod\, q)$, за винятком випадку $F_i = F_{j-i} = 1$. Застосуємо зменшення після якого отримуємо нові значення для a_j:

$$a_j = \begin{cases} \lambda, 2 \textit{ or } 3\lambda & \textit{if } F_j = 0 \textit{ or } F_{j-i} = 0, \\ 4\lambda - q \textit{ or } 5\lambda - q & \textit{if } F_j = 0 \textit{ or } F_{j-i} = 0. \end{cases}$$

Після зниження $a \bmod 2$, отримуємо значення 0 та 1, і більш того λ – парне, а *q* – непарне, після шага зниження отримаємо:

$$a_j \bmod 2 = \begin{cases} 0 & \textit{if } F_j = 0 \textit{ or } F_{j-i} = 0, \\ 1 & \textit{if } F_j = F_{j-i} = 1. \end{cases}$$

Тепер можемо визначити чітко $m'(e_i)$:

$$m'(e) = \sum_{j=0}^{N-1} \begin{pmatrix} 1 & F_i = F_{j-i} = 1 \\ 0 & otherwise \end{pmatrix} X^j. \qquad (15)$$

у полі якого маємо часткову інформацією про *F*:

$$F(e_i) = \sum_{j=0}^{N-1} \begin{pmatrix} 1 & \textit{if } m'(e_i)_j = 1 \\ & \textit{or } m'(e_i)_{i+j} = 1 \\ 0 & \textit{if } m'(e_i)_{j-i} = 1 \textit{ and } m'(e_i)_j \ne 1 \\ & \textit{or } m'(e_i)_{j+2i} = 1 \textit{ and } m'(e_i)_{j+i} \ne 1 \\ ? & otherwise \end{pmatrix} \qquad (16)$$

2.4. Обґрунтування вибору

Початкова множина ε відносно несильно знижена попередніми розрахунками. Функція атаки покликана повідомити зловмиснику, що він ідентифікує $m'(e_i)$, яке відповідає e_i в його базі. Проте, якщо нетривіально можливо те, що функція атаки – унікальна, то Б зможе вимагатиме перевірку, що він коректно визначив e_i. Відмітимо, якщо $e_i = \lambda + \lambda X^i$ і розшифруємо m', то це привело до альтернативної форми:

$$e_i^* = (\lambda + 2) + \lambda X^i, or\, \lambda + \lambda X^i \, or \ldots., \qquad (17)$$

або множина з $\lambda_1 + \lambda_2 X^i$ з цілими λ_1 та λ_2 у діапазоні $q/4$ і задовольняє $\lambda_1 + \lambda_2 > q/2$. Тому Б вибирає один з можливих e_i^* підраховує $T\left(m', e_i^*\right)$ для повідомлень m' так, щоб він вважав, що проводить розшифрування дійсного e_i^*, та що для представленого e_i^* знаходить кількість звернень за допомогою функції атаки. Якщо функція атаки вимірить та вирахує «подібну пару», то зловмисник сформулює припущення про m'. Проте дійсна пара функції атаки це простий збіг.

2.5. Практичні аспекти атаки за часом для $f=1+2F$

Далі на практиці буде показано як виконується аналогічна атака для секретного ключа виду $f=1+2F$, для цього буде використовуватися певний набір параметрів. Почнемо з опису використання геш-фукнції при вирахуванні *r*, та потім вирахуємо імовірність того, скільки раз цей процес використовує геш. Значення *r* (бінарного поліному з простих d_r) обчислюється з геш-функцій, через повторне використання деяких версій SHA. В результаті:

1. Маємо значення C, що задовольняє умові $2^c > N$. Також нехай $b = [c/8]$ та $n = [2^c/N]$, причому b це найменше ціле число таке, що b біт міститься принаймні у c біт. Таким чином n найменше множина N, що менше ніж $2c$.

2. Будемо використовувати SHA та поділимо вихід на найменші проміжки довжини b. В межах кожного такого проміжку, зберігається нижній регістр c біт та відкидається верхній регістр $8b–c$ біт. Конвертуємо нижній регістр c біт у прості числа $i_1, i_2, \cdot\cdot, i_t$.. На виході SHA складається з tb біт, де t просте число.

3. Створемо список $j_1, j_2, \cdot\ldots$ шляхом перебору i-х значень на кроці2. Якщо $i<n$ та $i \bmod N$ не знаходиться у списку, тто необхідно $i \bmod N$ додати до списку, в іншому випадку відкинути i. Продовжуємо список поки він містить j значення множини d_r. Якщо будь-яка точка не має i значення, то викликається SHA та створюється додаткове значення i як на кроці 2.

У порядку вирахування r, у алгоритмі описано необхідність створення списку $d_{r,}$, яке б задовольняло $0 \le i < N$. Кожного разу алгоритм використовує геш, він отримує t число, яке задовольняє $0 \le i < 2^c$. Звідси, імовірність того, що достатньо визвати SHA s раз, дорівнює імовірності, що випадкових st раз у межах $[0, 2^c)$ міститься принаймні значення d_r у діапазоні $[0, n)$, значення якої визначено модулем N. Звідси

$$\Pr ob\left(\textit{потрібних звернень } SHA\right) =$$

$$= \Pr ob\left(\begin{array}{c}\textit{випадкові } st \textit{ вибрані у диапазоні } [0,2^c) \\ \textit{включаючи не менше } d_r \textit{ у межах } [0,n) \\ \textit{узяті за модулем } N\end{array}\right).$$

Перша імовірність ближче до 0,5 це значить, що існує велика імовірність того що функція атаки унікальна.

ВИСНОВОК

В цій статті було показано атаку спеціального виду за часом, яка може розкрити секретний ключ NTRU. Ця атака можлива завдяки тому, що у розшифруванні різних шифр-текстів використовується різна кількість звернення до геш-функції. Хоча стаття було присвячена атаці на секретний ключ вид у $f = 1 + 2F$, це справедливо запропонувати і для ключів загального виду. Цей метод оснований на розшифруванні кількості звернень до геш-функції для кожного шифр-тексту. Тобто підраховується кількість звернень до геш-функції для кожного можливого шифр-тексту. Візьмемо максимальне значення та назвемо його $K_{max.}$ У випадку випадкового шифр-тексту, коли число звернень буде менше ніж K_{max}, то знадобитися додаткове використання геш-функції. Якщо шифр-тексту потрібно K – звернень до геш-функції, то $K_{max}–K$ буде потрібно додаткових звернень. Тому кількість звернень повинна бути однаковою для усіх шифр-текстів для попередження цієї атаки. Цей метод отримання кількості звернень на одне повідомлення, також по іншому можна назвати «доповненням». Для того, щоб попередити атаку спеціального виду потрібно збільшити стійкість самого алгоритму від атаки за часом, щоб кожен біт додавання впливав на кожен біт кількості звернень.

Література

[1] *J. Hoffstein, J. Pipher, J.H. Silverman* / NTRU: A new high speed public key cryptosystem, Algorithmic Number Theory (ANTS III)//, Portland, OR, June 1998, Lecture Notes in Computer Science 1423, J.P. Buhler (ed.), Springer-Verlag, Berlin, 1998, 267–288.

[2] *Jens Hermans, Frederik Vercauteren, Bart Preneel* / Speed Records for NTRU. // Topics in Cryptology - CT-RSA 2010, The Cryptographers Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings. Volume 5985 of Lecture Notes in Computer Science, pages 73-88, Springer, 2010.

[3] *Joseph H. Silverman, William Whyte* / NTRU Cryptosystems Technical Report Report 021, Version 1Timing Attacks on NTRUEncryption via Variation in the Number of Hash Calls // http://grouper.ieee.org/groups/1363/lattPK/ submissions /021NTRUTechReport-sha-timing.pdf

Надійшла до редколегії 20.03.2012

Бондаренко Михайло Федорович, член-кореспондент НАН України, лауреат державної премії України, доктор технічних наук, професор, ректор Харківського національного університету радіоелектроніки.

Балагура Дмитро Сергійович, фото та відомості про автора див. на с. 199.

Іваненко Дмитро Вікторович, аспірант кафедри БІТ ХНУРЕ. Область наукових інтересів: інформаційні технології, захист інформації, методи та засоби автентифікації.

УДК 621. 391:519.2:519.7

Атака специального вида на NTRU/ М.Ф. Бондаренко, Д.С. Балагура, Д.В. Иваненко // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 216–219.

Приводиться порядок шифрования и расшифрования алгоритма NTRU. Рассматривается атака специального вида по времени, демонстрируется возможность получить знание секретного ключа NTRU при известном количестве обращений до хеш-функции.

Ключевые слова: алгоритм NTRU, атака специального вида, шифрование, расшифрование, шифр-текст, хеш-функция, полином.

Библиогр.: 3 назв.

UDC 621. 391:519.2:519.7

Side channel attack on NTRU / M.F. Bondarenko, D.S. Balagura, D.V. Ivanenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 216–219.

The paper considers the order of encryption and decryption of the NTRU algorithm as well as it reviews the side channel attack and shows the possibility of obtaining knowledge about the secret key with a certain amount of hash-function calls.

Keywords: NTRU algorithm, side channel attack, encryption, decryption, cipher-text, hash function, polynomial.

Ref.: 3 items.

УДК 681.3.06

ПОРІВНЯЛЬНИЙ АНАЛІЗ КРИПТОГРАФІЧНИХ БІБЛІОТЕК З ВІДКРИТИМ КОДОМ ТА РЕКОМЕНДАЦІЇ З ЇХ ВИКОРИСТАННЯ

І.Ф. АУЛОВ, Ю.І. ГОРБЕНКО

Запропоновано методику, що дозволяє провести порівняльний аналіз криптографічних бібліотек за сукупністю критеріїв та показників. Наводяться результати порівняльного аналізу найпоширеніших криптографічних бібліотек: Crypto++, MIRACLE, OpenSSL, OpenPGP, Botan, GNU Crypto, CryptLib, NTL за запропонованою методикою. На основі вимірів, проведених відповідно до методології, яка враховує ефективність і відносну частоту використання примітивних криптографічних операцій, а також показників швидкості основних криптографічних перетворень виконується порівняння криптографічних бібліотек.

Ключові слова: криптографічна бібліотека, показники ефективності, порівняльний аналіз

ВСТУП

Сьогодні стало вже зрозуміло, що в комп'ютерних системах та мережах потрібно забезпечувати захист інформації не тільки в державних, банківських та комерційних установах, але й надавати основні послуги з захисту інформації звичайним користувачам.

Для звичайного користувача головним критерієм вибору засобу захисту персональної інформації є доступність цього засобу. Під доступністю засобу захисту розуміються умови розповсюдження, відсутність обмежень на використання та його ціна.

В сучасному світі серед користувачів найбільше розповсюдження отримали програмні засоби захисту інформації. Це пов'язано з низкою факторів, до яких відносяться: ціна (яка з об'єктивних причин нижче за ціну на аналогічні апаратні пристрої), кроссплатформеність (здатність програмного забезпечення функціонувати на різних платформах), простіша інтеграція в систему захисту, модульність та розширюваність (можливість в залежності від потреб захисту включати додаткові модулі розширяючи цим функціонал існуючих). Розглянуті аргументи на користь використання програмних засобів захисту, говорять про актуальність роботи з дослідження критеріїв та показників для порівняння програмних засобів захисту – криптографічних бібліотек.

Кожний користувач висуває до програмного засобу захисту свій унікальний набір критеріїв, з яких потім формуються різні показники. Унікальність набору критеріїв полягає не тільки в їх різному наборі, але й в ранжуванні цих критеріїв за ступенем важливості.

Метою цієї роботи є узагальнення вимог, що висуваються користувачами до криптографічних бібліотек та вироблення на їх основі сукупності критеріїв та показників для проведення порівняння криптографічних бібліотек.

Для проведення порівняльного аналізу у нашому дослідженні, було обрано найбільш поширені бібліотеки з відкритим кодом:

– NTL [5], OpenSSL [4], OpenPGP [1], GNU Crypto [3], CryptLib(Sleepycat) [7];

– умовно безкоштовні (BSD): Crypto++ [2], Botan [8];

– комерційні (AGPL): MIRACLE [6].

В процесі аналізу, було проведено оцінку придатності використання вищезгаданого програмного забезпечення бібліотеки для реалізації різноманітних криптосистем, з використанням ефективності реалізації примітивних операцій в якості основного критерію оцінки.

Порівняльний аналіз ефективності програмного забезпечення розглядається не тільки з точки зору ефективності реалізації криптографічного примітиву, а й як комплекс з великим числом змінних таких як, операційна система, процесор, об'єм пам'яті, вибір компілятора і його параметри оптимізації.

Також порівняльний аналіз включає в себе ряд вторинних критеріїв, таких як підтримка різноманітних криптографічних перетворень: асиметричні та симетричні криптоперетворення, функції гешування, реалізація протоколів, документація, простота використання і портативність.

Це дослідження ставить перед собою мету забезпечити розробників програмного забезпечення знаннями, необхідними, для того щоб зробити більш правильний вибір стосовно використання доступних бібліотек в своїх продуктах на основі набору критеріїв та показників.

1. КРИТЕРІЇ ТА ПОКАЗНИКИ ОЦІНКИ

Попередній аналіз показав, що більшість користувачів криптографічних бібліотек виділяють дві основні групи критеріїв: основні (базові) та вторинні. Під основними розуміються окремі критерії чи групи критеріїв, які є основними (найбільш важливими) для досягнення основної мети: обрання тієї чи іншої альтернативи. При цьому таким критерієм не можна нехтувати або вважати його не суттєвим, бо відмова від критерію призведе до отримання помилкового результату.

Під вторинними розуміються критерії, які не суттєво будуть впливати на остаточний результат, та можуть бути відкинуті в процесі аналізу.

В залежності від конкретної задачі набір основних критеріїв може змінюватися, та доповнюватися з групи вторинних.

Для вибору базових критеріїв при проведенні порівняльного аналізу бібліотек конкретизуємо кінцеву мету цього аналізу, як отримання бібліотеки, яка найбільш ефективно реалізує криптографічні перетворення.

Виходячи з мети аналізу до основних критеріїв буде відноситися:

– базові математичні операції, що використовуються в криптографічних алгоритмах;

– криптографічні алгоритми, що реалізовано;

– необхідні ресурси системи: пам'ять, процесорний час.

До вторинних критеріїв нами було віднесено:

– універсальність, розширюваність та переносиміть на інші платформи;

– доступність бібліотек;

– підтримка криптографічних та Інтернет протоколів;

– можливість проведення тестування.

Запропоновані критерії будемо оцінювати за наступними показниками:

– швидкість виконання базових математичних та криптографічних операцій (вимірюється в оп/с);

– швидкість виконання криптографічних функцій: шифрування, розшифрування, генерації ключем, гешування і т.д.(вимірюється в Мбіт/с);

– розмір загальносистемних параметрів та ключів (вимірюється в бітах);

– розмір пам'яті (вимірюється в Мб);

– процесорний час (такти процесора);

– підтримка різних програмних та апаратних платформ та компіляторів;

– можливості з оптимізації бібліотеки за рахунок налаштувань компілятору під певну платформу;

– можливість розпаралелювання;

– можливість експорту, необхідність отримання ліцензії та використання в комерційних додатках;

– наявність реалізованих протоколів;

– наявність функцій тестування, самотестування.

2. ЗАГАЛЬНА ХАРАКТЕРИСТИКА БІБЛІОТЕК ТА ПЛАТФОРМИ ДЛЯ ТЕСТУВАННЯ

Загальна характеристика бібліотек наводиться в табл. 1. В якості мови програмування в бібліотеках використовується C та C++. Більшість з цих бібліотеки можуть бути описані як ті, що реалізують криптографічні алгоритми: симетричного шифрування, функцій гешування, направленого шифрування, підпису. Бібліотеки NTL та MIRACLE представляють собою бібліотеки, що реалізують математику багато розрядної точності.

Такі бібліотеки, як Crypto++, OpenSSL, OpenPGP, GNU Crypto, CryptLib(Sleepycat) реалізують не тільки криптографічні алгоритми, але і мережеві протоколи: Kerberos, S/MIME, PGP, SSL/TLS, SSH.

Для аналізу ефективності реалізації алгоритмів було використано дві робочі станції з наступними характеристиками: Pentium IV 2,0 ГГц з 512 Мб ОЗУ та C2D 2,2 ГГц з 1 Гб ОЗУ. На машині Pentium IV та C2D було встановлено Windows XP32 (Cygwin). Бібліотеки були скомпільовані з використанням GNU C / C + + компілятору.

В табл. 2 наведено основні математичні операції, що реалізовано в бібліотеках для асиметричної криптографії. В табл. 3 наведено основні криптографічні алгоритми, які використовуються сьогодні або мають перспективу в застосуванні.

3. МЕТОДИКА ПОРІВНЯННЯ ЕФЕКТИВНОСТІ РЕАЛІЗАЦІЇ КРИПТОГРАФІЧНИХ АЛГОРИТМІВ

Для оцінки ефективності реалізації криптографічних алгоритмів для кожної бібліотеки, використаємо методику, яка наводиться в роботі [9].

Основні кроки методики порівняння:

1. Обирається множина бібліотек $LIB_0 ... LIB_{N-1}$.

2. Обирається множина алгоритмів за якими буде здійснюватися порівняння бібліотек $ALG_0 ... ALG_{M-1}$.

3. Обирається фіксований розмір вхідних параметрів для кожного алгоритму ALG_i .У випадку якщо алгоритм, який порівнюється приймає різні довжини параметрів, то потрібно обрати вектор довжин параметрів $PRM_0 ... PRM_{T-1}$.

4. Для кожної бібліотеки та кожного обраного алгоритму експериментально обчислюється

Таблиця 1

Загальна характеристика криптографічних бібліотек

№	Бібліотека	Тип бібліотеки	Ліцензія	Версія
1	NTL	Реалізує математику підвищеної точності	GNU GPL	5.3.1
2	MIRACLE	Реалізує математику підвищеної точності	AGPL	4.82
3	Botan	Криптографічна	BSD	1.10.1
4	Crypto++	Криптографічна	BSD	5.61
5	OpenSSL	Криптографічна, реалізує протокол SSL	GNU	0.9.7c
6	OpenPGP	Криптографічна, реалізую схему захищеної електронної пошти PGP	GNU	0.9
7	GNU Crypto	Криптографічна, реалізує протокол Kerberos	GNU	2.0.1
8	CryptLib	Криптографічна, протоколи Kerberos, S/MIME, PGP, SSL/TLS, SSH	GNU	3.4.1

Таблиця 2

Аналіз математичних операцій, що реалізовано в криптографічних бібліотеках

№	Бібліотека	Метод множення великих чисел	Метод зведення в ступінь за модулем	Метод знаходження GCD та xGCD	Каратцуби скалярного множення ЕК
1	NTL	Каратцуби	Зліва на право(≥2 -512), - Sliding window (≥512)	Узагальнений бінарний	-
2	MIRACLE	Звичайний/ Каратцуби	Sliding window (≥2)	Lehmer	wNAF-based interleaving
3	Botan	Каратцуби	Блочний метод (≥2	Узагальнений бінарний	Зліва на право
4	Crypto++	Каратцуби	З ліва на право(≥2 -32), блочний(≥32)	Евкліда	Simultaneous Sliding Window
5	OpenSSL	Звичайний/ Каратцуби	Sliding window (≥2)	Бінарний	wNAF-based interleaving
6	OpenPGP	Звичайний	Sliding window (≥2)	Бінарний	-
7	GNU Crypto	Каратцуби	З ліва на право(≥2 -32), блочний(≥32)	Узагальнений бінарний/ Lehmer	wNAF-based interleaving
8	CryptLib	Каратцуби	Sliding window (≥2)	Lehmer	wNAF-based interleaving

Таблиця 3

Криптографічні алгоритми, що реалізовано в криптографічних бібліотеках

№	Бібліотека	Симетрична криптографія			Асиметрична криптографія	Функції гешування
		Блокові шифри	Потокові шифри	Коди аутентифікації		
1	NTL	-	-		-	-
2	MIRACLE	Rijndael AES	-	-	**Підпис:** RSA, DSA, ECDSA, ECGDSA, ECKCDSA **НШ:** RSA	SHA-1, SHA-2
3	Botan	Rijndael AES, Serpent, Twofish, TDES, GOST 28147	ARC4, Salsa20/ XSalsa20, Turing	HMAC, CMAC (aka OMAC1), CBC-MAC, ANSI X9.19, DES-MAC	**Підпис:** RSA, DSA, ECDSA, GOST 34.10-2001, Nyberg-Rueppel, **НШ:** RSA, ElGamal **ВК:** Diffie-Hellman, ECDH	SHA-1, SHA-2, Skein-512, Keccak, Whirlpool, Tiger, GOST 34.11
4	Crypto++	Rijndael AES, Twofish, Serpent, TDES	Panama, Sosemanuk, Salsa20, XSalsa20	VMAC, HMAC, GMAC (GCM), CMAC, CBC-MAC, DMAC	**Підпис:** ECDSA, RSA, DSA, ECNR, **НШ:** RSA, ElGamal, Nyberg-Rueppel, **ВК:** DH, DH2, ECDH	SHA-1, SHA-2, Tiger, WHIRLPOOL, RIPEMD-256, RIPEMD-320
5	OpenSSL	Rijndael AES, GOST 28147 TDES	-	GOST 28147-89 MAC, HMAC	**Підпис:** ECDSA, RSA, DSA **НШ:** RSA, ElGamal, **ВК:** DH, ECDH	SHA1
6	OpenPGP	Rijndael AES, Twofish, TDES	-	-	**Підпис:** RSA, DSA **НШ:** RSA, ElGamal	SHA-1, SHA-2
7	GNU Crypto	Rijndael AES, Twofish, Serpent, TDES	-	HMAC, UMAC	**Підпис:** RSA, DSS **НШ:** RSA, ElGamal **ВК:** DH	SHA-1, SHA-2
8	CryptLib	Rijndael AES, Twofish, Serpent, TDES	-	HMAC	**Підпис:** ECDSA, RSA, DSA **НШ:** RSA, ElGamal, **ВК:** DH, ECDH	SHA-1, SHA-2

вектор $LIB_k^{ALG_i(PRM_t)}$, i={0...M-1}, k={0...N-1}, t={0...T-1} значень часу виконання довжиною L.

5. Для кожної бібліотеки LIB_k та кожного алгоритму $ALG_i(PRM_t)$ з вектором параметрів PRM_t обирається мінімальне значення $MIN(LIB_k^{ALG_i(PRM_t)})$ з вектору $LIB_k^{ALG_i(PRM_t)}$.

6. Використовуючи $MIN(LIB_k^{ALG_i(PRM_t)})$ для кожного алгоритму $ALG_i(PRM_t)$ формується вектор мінімальних значень для бібліотек розміром N: $MIN_j(LIB_k^{ALG_i(PRM_t)})$, j={0, N-1}.

7. З цього вектору обирається мінімальне значення $MIN(MIN_j(LIB_k^{ALG_i(PRM_t)}))$, та за ним нормується весь вектор значень.

8. Далі виконується обчислення загального значення для кожного алгоритм кожної бібліотеки $RANK(LIB_k^{ALG_i})$ для всіх значень параметрів PRM_t:

$$RANK(LIB_k^{ALG_i}) = \sqrt[T-1]{\prod_{t=0}^{T-1} \frac{MIN_j(LIB_k^{ALG_i(PRM_t)})}{MIN(MIN_j(LIB_k^{ALG_i(PRM_t)}))}}$$

9. Загальний результат для бібліотеки LIB_k, обчислюється як:

$$RANK(LIB_k) = \sqrt[M]{\prod_{i=0}^{M-1} LIB_k^{ALG_i}}.$$

Порівняння бібліотек за значенням $RANK(LIB_k)$ дозволяє отримати найбільш ефективну бібліотеку за набором алгоритмів ALG_i, та розмірів вхідних параметрів PRM_t, чим менше це значення тим більш ефективною вважається бібліотека.

4. РЕЗУЛЬТАТИ ПОРІВНЯЛЬНОГО АНАЛІЗУ ЕФЕКТИВНОСТІ РЕАЛІЗАЦІЇ КРИПТОГРАФІЧНИХ АЛГОРИТМІВ

Для порівняння ефективності реалізації криптографічних алгоритмів за методикою, наведеною в пункті 3, в бібліотеках було обрану наступну множину функцій:

– операцій множення великих чисел (MUL);

– зведення до ступеня за модулем (POWMOD);

– знаходження зворотнього елементу за модулем двох чисел (xGCD);

– ECMUL – множення точки ЕК на скаляр над полем F2m.

Для кожного алгоритму було виконано 100 тестів. В результаті були отримані відповідні значення RANK (табл. 4) для кожної з бібліотек та функцій відповідно.

Для порівняння ефективності реалізації крипто алгоритмів за показником швидкості було обрано:

– для симетричних шифрів AES-192(Rijnael) та TDES;

– для підпису RSA (2048), та ECDSA (F2m=283);

– код-аутентифікації повідомлення HMAC (SHA-1) блок даних 256 байт;

– функцію гешування SHA-1, блок даних 256.

Отримані результати наведені в табл. 4. В табл. 5 наведені реальні значення швидкості розглянутих криптографічних алгоритмів.

Таблиця 4

Результати оцінки ефективності реалізації криптографічних алгоритмів

№	Бібліотека	*RANK(ALG$_i$)*				AES-192	TDES	RSA (2048)		ECDSA (F2m=283)		HMAC SHA-1	SHA-1 ($l_{бл}$=256)	RANK
		MUL	POWMOD	xGCD	ECMUL			*sign*	*verf*	*sign*	*verf*			
1	NTL	1,01	1,18	1,0	-	-	-	-	-	-	-	-	-	1,06
2	MIRACLE	3,58	2,62	3,15	1	1,00	-	2,29	1,72	2,02	2,76	-	1,12	1,40
3	Botan	3,41	5,21	1,09	1,42	2,16	1,98	1,00	1,00	1,60	1,95	1,15	1,13	1,67
4	Crypto++	4,49	5,04	16,82	4,35	2,68	1,90	1,12	1,15	1,00	1,00	1,05	1,07	2,19
5	OpenSSL	2,80	2,43	12,49	1,15	4,49	3,39	1,68	1,51	1,85	2,53	1,00	1,00	2,26
6	OpenPGP	2,87	2,31	3,11	1,41	1,12	1,37	2,35	1,73	1,54	2,00	-	1,06	1,79
7	GNU Crypto	1,0	1,0	1,01	1,24	1,38	1,00	2,71	1,80	1,75	2,13	1,06	1,08	1,35
8	CryptLib	5,25	4,17	8,59	1,7	1,03	1,47	1,09	1,08	1,07	2,05	1,02	1,06	1,82

Таблиця 5

Швидкість розглянутих криптографічних алгоритмів

№	Бібліотека	AES-192(Rijnael)	TDES	RSA(2048)		ECDSA (F2m=283)		HMAC	SHA-1
				sign	verf	sign	verf		
		KB/s	KB/s	op/s	op/s	op/s	op/s	MB/s	MB/s
1	NTL	-	-	-	-	-	-	-	-
2	MIRACLE	10,125	-	78	2412	303	105	-	160
3	Botan	21,917	6,979	34	1404	240	74	161	162
4	Crypto++	27,111	6,702	38	1612	150	38	147	153
5	OpenSSL	45,461	11,94	57	2120	278	96	140	143
6	OpenPGP	11,311	4,845	80	2430	231	76	-	151
7	GNU Crypto	13,925	3,525	92	2531	263	81	149	155
8	CryptLib	10,398	5,195	37	1512	161	78	143	151

ВИСНОВКИ

За результатами порівняльного аналізу ефективності криптографічних бібліотек можна зробити наступні висновки:

– бібліотеки NTL та GNU Crypto показали кращі показники за операціями з великими цілими числами;

– кращим вибором для розробників програмного забезпечення, що використовує арифметику великих цілих чисел, з точки зору швидкості перетворень буде бібліотека GNU Crypto;

– якщо орієнтуватися на кількість зусиль розробника, які необхідно прикласти при розробці криптографічного алгоритму, а також на переносимість на інші платформи, то слід звернути увагу на бібліотеки MIRACLE та OpenSSL;

– в процесі вибору криптографічної бібліотеки розробник важливим фактором є апаратна та програмна платформа на якій буде використовуватися бібліотека, тому слід приділяти увагу налаштуванням компілятора та бібліотеки для кожної платформи;

– при використанні значення RANK для вибору криптографічної бібліотеки слід також звертати увагу на конкретні показники швидкості для криптографічних перетворень, які будуть використовуватися, бо на значення RANK впливають сукупність усіх показників.

Література

[1] OpenPGP: The Open Source toolkit for PGP http://openpgp.nominet.org.uk/

[2] Crypto++ Library 5.1: a Free C++ Class Library of Cryptographic schemes http://www.eskimo.com/weidai/cryptlib.html

[3] The GNU Crypto Library http://www.swox.com/gmp/

[4] OpenSSL: The Open Source toolkit for SSL/TLS http://www.openssl.org/

[5] NTL: A Library for doing Number Theory http://www.shoup.net/ntl/

[6] Shamus Software Ltd MIRACL http://indigo.ie/mscott/

[7] CryptLib: Security toolkit http://www.cs.auckland.ac.nz/~pgut001/cryptlib/

[8] Botan: Security toolkit http://botan.randombit.net/download.html

[9] Ashraf Abusharekh, Kris Kaj, Comparative Analysis of Software Libraries for Public Key Cryptography. — CRC-Press, 2007 – 18 p.

Надійшла до редколегії 28.03.2012

Аулов Іван Федорович, магістрант кафедри БІТ ХНУРЕ. Область наукових інтересів: дослідження принципів побудови, розгортання і аналізу стійкості асиметричних криптографічних систем.

Горбенко Юрій Іванович, фото та відомості про автора див. на с. 187.

УДК 681.3.06

Сравнительный анализ криптографических библиотек с открытым кодом и рекомендации по их использованию / И.Ф. Аулов, Ю.И. Горбенко // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 220–224.

Предложено методику, которая позволяет провести сравнительный анализ криптографических библиотек за совокупностью критерием и показателей. Приводятся результаты сравнительного анализа наиболее распространенных криптографических библиотек: Crypto++, MIRACLE, OpenSSL, OpenPGP, Botan, GNU Crypto, CryptLib, NTL по предложенной методике. На основе измерений, проведенных соответственно методологии, которая учитывает эффективность и относительную частоту использования примитивных криптографических операций, а также показателей скорости основных криптографических преобразований выполняется сравнение криптографических библиотек.

Ключевые слова: криптографическая библиотека, показатель эффективности, сравнительный анализ.

Табл. 05. Библиогр.: 09 назв.

UDC 681.3.06

Comparative analysis of open source cryptographic libraries and recommendations for their use / I.F. Aulov, Yu.I. Gorbenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 220–224.

The paper proposes a technique which allows to make a comparative analysis of cryptographic libraries for the collection of criteria and indicators. Results of the comparative analysis of the most common cryptographic libraries: Crypto + +, MIRACLE, OpenSSL, OpenPGP, Botan, GNU Crypto, CryptLib, NTL on the proposed methodology are presented. The comparative analysis is implemented on the basis of the measurements taken, according to a methodology that takes into account the effectiveness and relative frequency of using primitive cryptographic operations and key indicators of the speed of the main cryptographic transformations.

Keywords: cryptographic library, performance indicator, comparative analysis.

Tab. 05. Ref.: 09 items.

УДК 681.3.06

КРИВЫЕ ЭДВАРДСА ПОЧТИ ПРОСТОГО ПОРЯДКА НАД РАСШИРЕНИЯМИ МАЛЫХ ПРОСТЫХ ПОЛЕЙ

А.В. БЕССАЛОВ, А.И. ГУРЬЯНОВ, А.А. ДИХТЕНКО

В задаче поиска криптостойких кривых Эдвардса предложен подход, состоящий в построении кривой минимального порядка 4 над малыми полями F_5 и F_7 с последующим простым расширением этих полей. Найдены порядки 5 кривых, которые можно рекомендовать в будущих стандартах и проектируемых криптосистемах.

Ключевые слова: эллиптические кривые, кривые Эдвардса, порядок кривой, порядок точки, простое поле, расширенное поле.

ВВЕДЕНИЕ

Традиционные асимметричные криптосистемы на эллиптических кривых свыше десятилетия успешно применяются на основе действующих национальных и международных стандартов [5]. Вечные поиски более совершенных алгоритмов привели в последние годы к замечательной альтернативе канонической формы кривых – кривым в форме Эдвардса [1 – 4]. Главные их достоинства: рекордная производительность и простота программирования. Вместе с тем кривые Эдвардса пока не стандартизированы, для них необходимо инициировать поиск кривых с почти простым порядком $N_E = 4n$, где n – простое число. Минимальный кофактор 4 в порядке кривой связан с наличием в любой такой кривой двух точек 4-го порядка. Нетрудно ограничением на параметр d кривой исключить точки 8-го порядка [3], но в общем случае для поиска подходящих кривых Эдвардса придется адаптировать известные алгоритмы SEA или Satoh [5].

В настоящей работе предлагается наиболее простой путь нахождения кривой Эдвардса почти простого порядка $4n$. Наподобие с кривыми Коблитца над полями характеристики 2, мы предлагаем найти две кривые Эдвардса минимального порядка $N_{E1} = 4$ над малыми простыми полями F_5 и F_7, после чего найти порядки этих кривых над расширениями степени m этих полей с последующим отбором при простых m подходящего почти простого порядка $4n$. В результате нами были определены несколько кривых в области криптографических приложений.

ПОИСК КРИВЫХ ЕДВАРДСА ПОЧТИ ПРОСТОГО ПОРЯДКА И РЕЗУЛЬТАТЫ

Форма кривых Эдвардса над конечными полями характеристики $p > 3$ имеет вид

$$x^2 + y^2 = c^2\left(1 + \tilde{d}x^2y^2\right),$$
$$\tilde{d} = c^{-4}d,\ \tilde{d}\left(1 - \tilde{d}c^4\right) \neq 0,\ \tilde{d} \neq A^2.$$

Здесь все множество различных значений параметра c дает изоморфные кривые, поэтому можно принять $c = 1$, $\tilde{d} = d$, тогда различные кривые Эдвардса определяются лишь одним параметром d в уравнении

$$x^2 + y^2 = \left(1 + dx^2y^2\right),$$
$$d(1 - d) \neq 0,\ d \neq A^2. \quad (1)$$

Пусть кривая определена над полем F_5, здесь допустимыми значениями параметра d являются квадратичные невычеты 2 и 3. Они являются мультипликативно обратными, поэтому образуют пару кривых кручения. Границы Хассе $p + 1 \pm 2\sqrt{p}$ при $p = 5$ лежат в интервале 2...10, в пределах которого для кривых Эдвардса допустимы лишь 2 значения порядка N_E кривой, равные 4 и 8. Согласно утверждению 1 в [3] точка 8-го порядка существует, если $1 - d$ – квадратичный вычет, и не существует в противном случае. При $d = 2$ значение $(1 - d) = 4 \bmod 5$ – квадратичный вычет, и соответствующая кривая имеет порядок 8. При $d = 3$ значение $(1 - d) = 3 \bmod 5$ – квадратичный невычет, и соответствующая кривая имеет порядок 4. Она содержит обязательные 4 точки всех кривых Эдвардса $(0, \pm 1), (\pm 1, 0)$ при $c = 1$.

Итак, мы принимаем $d = 3$, тогда из $N_{E1} = p + 1 - t_1 = 4$ след уравнения Фробениуса $t_1 = 2$. Рассчитаем порядки кривых над расширениями F_p^m по известной формуле [5]

$$N_{Em} = p^m + 1 - t_m, \quad (2)$$

где для определения параметра t_m воспользуемся рекуррентной зависимостью

$$t_m = t_1\, t_{m-1} - pt_{m-2},$$
$$m = 2, 3, \ldots,\ t_0 = 2. \quad (3)$$

Результаты расчетов по формулам (2), (3) с отбором простых значений $n = N_{Em}/4$ приведены в табл. 1. Во второй колонке таблицы даны округленные значения для длины модуля поля $m_b = m \log p / \log 2$ в битах. Тестирование числа n на простоту с помощью алгоритма Миллера-Рабина осуществлялось специальной прикладной программой.

В границах Хассе имеется еще одна кривая с минимальным порядком $N_{E1} = p + 1 - t_1 = 4$ при $p = 7$ и $t_1 = 4$. Она также имеет параметр $d = 3$, который является квадратичным невычетом в поле F_7, причем $1 - d = 5$ – тоже невычет. Почти простые порядки этой кривой над расширениями F_7^m, рассчитанные с помощью (2), (3), даны в табл. 2.

$p = 5$ **Таблица 1**

m	m_b	$n = N_{Em}/4$
3	7	37
5	11	761
17	39	190734426721
47	109	17763568394002504911187090255831 7
53	123	27755575615628913519432138978855094 01
181	420	81566305849981556583878676365706844446264553225862081846982955622470058935583394181280598 16686403639171062258340162734855132 41
227	527	11591269220898191830411672692336373479273639933618096882665747059117441687798840670250687 80602938200802665596049849635508726680050691849860699590321446843229 17
353	819	13625471488026082303712171891991388314389109549794181122960160293908508251985766836112118 0 27927542086233890704552817681219819158519647915156383473783742883700653042365583720331179 91089062162100200930469700901559446602358040911814920317902577678401

$p = 7$ **Таблица 2**

m	m_b	$n = N_{Em}/4$
5	14	4261
7	19	205759
17	47	58157621574673
43	120	54595359399794914922465326744889728 3
47	132	13108345791890759086345450437985587821 83
127	356	53136273114200417711082595776474056083298454184099962702599160022401657332487956399341333 796788130398754359
223	626	71581852226941622933299737411659197932981104415173763451661527073195598927924017803839396 07571148856774258554873765716518606020812820444545621959754591269503845751314733544709671 6383526039

Приходится констатировать, что наши априорные ожидания достаточно большого числа приемлемых для криптографии кривых Эдвардса над расширениями малых простых полей характеристики $p > 3$ не подтвердились. Как следует из таблиц 1 и 2, в границах стандартных требований к порядку генератора криптосистемы и близким к нему расширением $2^{mb}\,(m_b \cong 180\ldots600)$ мы нашли всего 3 кривые Эдвардса: 2 кривые над полем $F_5{}^m$ со степенями $m = 181$ и $m = 227$, и одну кривую над полем $F_7{}^m$ со степенью $m = 127$. К ним, правда, можно добавить еще 2 кривые с завышенным уровнем стойкости и значением $m_b > 600$ (со временем он перестанет быть завышенным).

Следует заключить, что найденные кривые с минимальным и простым значением параметра $d = 3$ обеспечат при заданной стойкости наивысшую скорость вычислений групповых операций. В операции сложения разных точек мы экономим на одной полевой операции умножения 1U на параметр кривой [4], так как умножение на 3 заменяется трехкратным сложением в поле, т.е. практически бесплатной операцией. Арифметика вычислений в расширениях малых полей часто эффективней арифметики в простых полях большой характеристики. Полагаем, что найденные кривые можно рекомендовать как для проектов будущих стандартов, так, возможно, и для использования в криптопротоколах уже сегодня.

Литература

[1] *Edwards H.M.* A normal form for elliptic curves. Bulletin of the American Mathematical Society, Volume 44, Number 3, July 2007, Pages 393-422.

[2] *Bernstein Daniel J., Lange Tanja.* Faster addition and doubling on elliptic curves. IST Programme under Contract IST–2002–507932 ECRYPT, 2007, PP. 1-20.

[3] *Бессалов А.В.* Число изоморфизмов и пар кручения кривых Эдвардса над простым полем. Радиотехника, вып. 167, 2011. С. 203-208.

[4] *Бессалов А.В., Дихтенко А.А., Третьяков Д.Б.* Сравнительная оценка быстродействия канонических эллиптических кривых и кривых в форме Эдвардса над конечным полем. Сучасний захист інформації, №4, 2011. – с.33 – 36.

[5] *Бессалов А.В., Телиженко А.Б.* Криптосистемы на эллиптических кривых: Учеб. пособие. – К.: ІВЦ «Політехніка», 2004. – 224 с.

Поступила в редколлегию 30.03.2012

Бессалов Анатолий Владимирович, доктор технических наук, профессор, профессор кафедры ММЗИ ФТИ НТУУ «КПИ». Область научных интересов: криптография, теория корректирующего кодирования.

Гурьянов Александр Игоревич, выпускник лицея №38 им. В.М. Молчанова гор. Киева. Область научных интересов: асимметричная криптография.

Дихтенко Алиса Анатольевна, аспирант кафедры теории упругости и вычислительной математики Донецкого национального университета. Область научных интересов: асимметричная криптография.

УДК 681.3.06

Криві Едвардса майже простого порядку над розширеннями малих простих полів / А.В. Бессалов, О.І. Гурьянов, А.А. Діхтенко // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 225–227.

У задачі пошука криптостійких кривих Едвардса запропоновано підхід, що полягає в побудові кривої мінімального порядку 4 над малими полями F_5 и F_7 з наступним простим розширенням цих полів. Знайдені порядки 5 кривих, які можна рекомендувати в майбутніх стандартах і проектуємих криптосистемах.

Ключеві слова: еліптичні криві, криві Едвардса, порядок кривої, порядок точки, просте поле, розширене поле.

Табл. 02. Бібліогр.: 5 найм.

UDC 681.3.06

Edwards curves of almost prime order over extensions of small prime fields / A.V. Bessalov, A.I. Gur'yanov, A.A. Dikhtenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 225–227.

The paper suggests an approach to the problem of seaching for Edwards curves acceptable to cryptography which consists in constructing a curve of minimum order 4 over the small fields F_5 and F_7 with a consequent prime extension of the said fields. The orders of five curves are found which can be recommended for use in future standards and cryptosystems under design.

Keywords: elliptic curves, Edwards curves, curve order, point order, prime field, extension field.

Tab. 02. Ref.: 5 items.

УДК 621.391

АНАЛІЗ, ПОРІВНЯННЯ ТА ОСОБЛИВОСТІ АРХІТЕКТУРИ ФУНКЦІЇ ГЕШУВАННЯ BLAKE ПРОЕКТУ SHA-3

Є.Ю. КУТЯ, І.Д. ГОРБЕНКО

Наводяться порівняння функції гешування BLAKE відносно інших чотирьох фіналістів проекту SHA-3. Наведені результати аналізу архітектури геш-функції BLAKE оцінка результатів швидкодії реалізації алгоритму.

Ключові слова: алгоритм гешування, колізія, швидкодія, стійкість, функція стиску, початковий стан, однонаправленість, ітеративна структура.

ВСТУП

У зв'язку з розвитком інформаційно-телекомунікаційних систем, після останніх досягнень в сфері криптоаналізу геш-функцій, виникла проблема створення нових, більш захищених функцій гешування. Для прийняття нового стандарту функцій гешування було вирішено провести конкурс геш-функйій SHA-3 Competition, ініційований Національним інститутом стандартов і технологій США (NIST). На конкурс було представлено 64 конкурсанта. Після трьох раундів та декількох років аналізу та порівняння до фіналу вийшли 5 алгоритмів. Європейське криптографічне співтовариство проводило параллельний конкурс на базі проекту Ecrypt. Переможцем цього конкурсу стала геш функція BLAKE, створена командою зі Швейцарії. Задача аналізу та порівняння геш-функцій, які пройшли до фіналу конкурсу, та аналіз переваг архітектури та властивостей геш-функції BLAKE залишається актуальною на сьогоднішній день через те, що міжнародний конкурс ще не закінчився, і кожен за кандидатів може бути внесений до національного стандарту фукцій гешування США. Доцільно більш глибоко розглянути архітектуру саме алгоритму BLAKE, як найбільш ймовірного переможця, а значить, алгоритм, який буде реалізований та впроваджений у всі сучасні інформаційні системи.

1. АЛГОРИТМ BLAKE

В цьому розділі розглядається алгоритм BLAKE: історія, архітектура та криптографічний зміст.

1.1. Основа для створення

Функція BLAKE розроблена командою з чотирьох незалежних розробниців зі Швейцарії. Головним розробником вважається Жан-Філіппе Аумассон (Jean-Philippe Aumasson), видатний криптограф та дослідник комп'ютерних наук. Алгоритм BLAKE базується на "ChaCha", версії поточного шифру Salsa20, надійність якого вже була досліджена та описана Аумассоном. Хоча геш принципово відрізняється від двонаправленої природи шифрів, загальними залишаються операції перетворення, до того ж вони передюбачають однакові криптографічні вимоги. Початкова специфікація алгоритму була подана до NIST у 2008 році, але була вдостконалена протягом 2010 року[1].

1.2.Структура Алгоритму

Розробники запропонували чотири офіційні варіанти BLAKE, які відрізняються вихідним розміром геш-значення, згідно до вимог конкурсу. Офіційними вважаються варіанти з вихідним розміром геш-значення 224, 256, 384 та 512 біт (як зазначено у назві BLAKE-224, BLAKE-256 і т.д.). Розміри, що підтримуються, відповідають розмірам SHA-2 з метою підвищення сумісності. Представлені чотири версії також мають відмінності у розмірі оброблюваного повідомлення та максимальній вхідній довжині повідомлення.

BLAKE – це ітеративний алгоритм, що базується на добре-відомій та визнаній ітеративній структурі Hash Iterative Framework (HAIFA). Первісний геш, h^0, загружається з передвизначеного вектору ініціалізації. Розробники BLAKE вирішили використати вектори ініціалізації, що використовуються в SHA-2, з причини її сильних псевдо-випадкових властивостей. Далі вхідне повідомлення розбивається на N еквівалентнорівних блоків (з розширенням до розміру блоку за потребою) та проходить разом із нульовим геш-значенням крізь функцію стиску. Результатом є нове геш-значення. Далі, цей процес повторюється до тих пір поки все повідомлення не буде стиснуте.

Останній результат функції стиску приймається як фінальний геш повідомлення. Ця структура зображена на рис. 1. Головною перевагою структури HAIFA є те, що довге повідомлення може бути гешовано, нібито у потоці, без необхідності мати все повідомлення у доступній пам'яті водночас.

$$h^0 \leftarrow IV$$
$$\textbf{for } i = 0, \ldots, N-1$$
$$\qquad h^{i+1} \leftarrow \textbf{compress}(h^i, m^i, s, \ell^i)$$
$$\textbf{return } h^N$$

Рис. 1. Використання конструкції HAIFA у BLAKE

Функція стиску є головною особливістю структури HAIFA, але її детальна функціональність визначена специфікаціями конкретного алгоритму гешування. Функція стиску BLAKE

визначена у наступній секції. Важливо зазначети, що в функцію передається значення «таймеру»(так його назвали розробники), яке визначає кількість вже оброблених біт. Такий метод допомагає забеспечити те, що ідентичні блоки у ріних частинах вхідного повідомлення будуть мати різні геш-значення. Розробники BLAKE також вирішили включити змінну «сіль» (далі - *salt*) до функції стиску, що забезпечує більшу захищеність від коллізійних атак. Додавання *salt* не є вимогою висунутою NIST, тому це значення встановлюється в нуль, якщо не впроваджено або не визначено користувачем.

1.3. BLAKE-256

В цьому розділі будуть розглянуті деталі версії алгоритму BLAKE-256. Для спощення, три другі версії алгоритму будуть розглядатись тільки в цілях створення акценту на відмінностях між ними.

Функція BLAKE-256 виробляє 256-біт геш-значення, яке розуміється як вісім 32-бітних слів. Вона приймає на вхід повідомлення довжиною $0 \le l \le 264$ біт. Повідомлення доповнюється одиничним бітом, після цього доповнюється послідовністю з нульових бітів та має 64-бітне представлення довжини *l*. Число нульових бітів визначено таким чином, що загальна довжина потоку є поєднанням 512-бітних блоків. Це дозволяє використовувати 512-бітні блокі як вхідні в функцію стіску для обробки.

Функція стиску приймає на вхід чотири параметри: 256 біт поточного значення геш (h^i), 512-бітній блок вхідного повідоплення (m^i), 128-біт *salt* (*s*) та кумулятівній лічільнік 64-бітних повідомлень (l^i). Треба зазначити, що l^i не включає біти розширення (паддінгу), тобто 100-бітне повідомлення буде мати l^i=100 , а не 512. Якщо кінцевий блок складається тільки з бітів розширення, такий випадок вважається особливим та l^N передається як 0, для впевненності відмінності від попереднього значення ітерації.

Початковий геш h0 використаний з SHA-2, як вже зазначалось вище. Він називається Вектором Ініціалізації (IV) та зображений на рис. 2, як вісім шістнадцятирічних слів.

$$IV_0 = 6A09E667 \quad IV_1 = BB67AE85$$
$$IV_2 = 3C6EF372 \quad IV_3 = A54FF53A$$
$$IV_4 = 510E527F \quad IV_5 = 9B05688C$$
$$IV_6 = 1F83D9AB \quad IV_7 = 5BE0CD19$$

Рис. 2. Вектор ініціалізації, який використовується в BLAKE-256

$$\begin{pmatrix} h_0 & h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 & h_7 \\ s_0 \oplus c_0 & s_1 \oplus c_1 & s_2 \oplus c_2 & s_3 \oplus c_3 \\ t_0 \oplus c_4 & t_0 \oplus c_5 & t_1 \oplus c_6 & t_1 \oplus c_7 \end{pmatrix}$$

Рис. 3. Початковий стан функції стиску

Всередині функції стиску, 512-бітний стан *v* зберігається та оброблюється як матриця 4x4 32-бітних слів. Цей стан ініціалізується з поточного геш-значення, значення *salt*, значення таймеру *t* та 256-бітної констнанти *c*, як зображено на рис. 3 (значення l^i що передається всередину функції називається таймером, який відрізняється від значення довжини всередині функції стиску). Значення константи *c*, прямо взяті з шістнадцятирічного представлення π, обранного для такої ітераційної структури (рис. 4).

Після ініціалізації матриці стану, вона ітеративно проходить 14 раундів обробки. Розробники геш-функції BLAKE обрали модель захисту, яка використовує невелику кількість складних раундів, в той час як велика кількість дослідників віддають перевагу великій кількості менш складних раундів.

$$c_0 = 243F6A88 \quad c_1 = 85A308D3$$
$$c_2 = 13198A2E \quad c_3 = 03707344$$
$$c_4 = A4093822 \quad c_5 = 299F31D0$$
$$c_6 = 082EFA98 \quad c_7 = EC4E6C89$$
$$c_8 = 452821E6 \quad c_9 = 38D01377$$
$$c_{10} = BE5466CF \quad c_{11} = 34E90C6C$$
$$c_{12} = C0AC29B7 \quad c_{13} = C97C50DD$$
$$c_{14} = 3F84D5B5 \quad c_{15} = B5470917$$

Рис. 4. Константи *c*, алгоритму BLAKE, отримані з π

Обидві парадигми мають своїх прихильників, але чи є одна з парадигм більше захищена ніж друга, доведено не було. («Захищена» в цьому контексті означає складність інвертування, тобто визначення вхідних даних, використовуючи вихідні).

Кожний раунд складається з восьми перетворень стану, які обозначають як G0-G7. Ці перетворення відповідаються за змішування даних (змінення даних) та дифузію бітів (розсіювання існуючих даних) по всім даним.

Кожний з раундів перетворює тільки 4 з 16 слів стану, які обозначаються як *a*, *b*, *c* та *d*. Перетворення (включаючи додавання, інверсії, виключення або XOR-операції) представлені на рис. 5.

$$\begin{aligned} a &\leftarrow a + b + (m_{\sigma_r(2i)} \oplus c_{\sigma_r(2i+1)}) \\ d &\leftarrow (d \oplus a) \ggg 16 \\ c &\leftarrow c + d \\ b &\leftarrow (b \oplus c) \ggg 12 \\ a &\leftarrow a + b + (m_{\sigma_r(2i+1)} \oplus c_{\sigma_r(2i)}) \\ d &\leftarrow (d \oplus a) \ggg 8 \\ c &\leftarrow c + d \\ b &\leftarrow (b \oplus c) \ggg 7 \end{aligned}$$

Рис. 5. Головне перетворення G

Головна функція перетворення G відображена на рис. 6 у вигляді візуальної схеми перетворень. Індекс σ_r відповідає одному з десяти перетворень чисел з 0 до 15, індексовані номером раунду *r* за модулем 10. Ця функція має вирішуюче значення

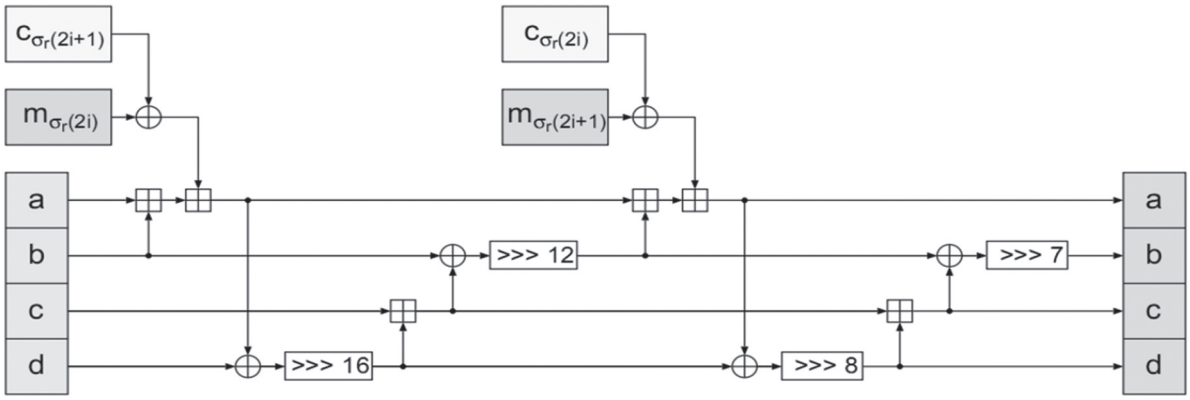


Рис. 6. Візуальне представлення функції G

для відповідного розсіювіання вхідних даних. Таблиця перестановок цього перетворення представлена на рис. 7.

Стани слова, по якому кожне перетворення функції G діє, були спеціально зформовані для підвищення ефективності. Перші чотири оперують над незалежними стовбцями та можуть виповнюватись параллельно. Останні чотирі оперують над незалежними діагональними значеннями, які також можуть бути виконані у паралельному режимі.

Таким чином раунд може бути зображений у вигляді двох великих операцій, ніж розглядати це як вісім послідовних операцій. Першаоперація названа як діагональний крок, а друга – вертикальний. Це істотно зменшує час обчислення одного раунду на 75% у аппаратній та деяких програмних реалізаціях. Таке розпараллелювання представлене на рис. 8, зображаючи, які слова обробляються кожною операцією.

σ_0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
σ_1	14	10	4	8	9	15	13	6	1	12	0	2	11	7	5	3
σ_2	11	8	12	0	5	2	15	13	10	14	3	6	7	1	9	4
σ_3	7	9	3	1	13	12	11	14	2	6	5	10	4	0	15	8
σ_4	9	0	5	7	2	4	10	15	14	1	11	12	6	8	3	13
σ_5	2	12	6	10	0	11	8	3	4	13	7	5	15	14	1	9
σ_6	12	5	1	15	14	13	4	10	0	7	6	3	9	2	8	11
σ_7	13	11	7	14	12	1	3	9	5	0	15	4	8	6	2	10
σ_8	6	15	14	9	11	3	0	8	12	2	13	7	1	4	10	5
σ_9	10	2	8	4	7	6	1	5	15	11	9	14	3	12	13	0

Рис. 7. Десять перетворень визначені для BLAKE

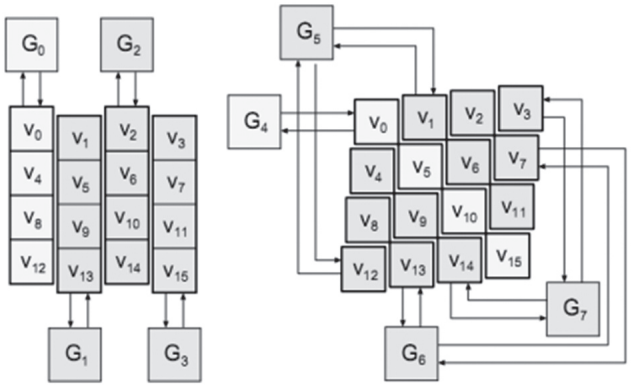


Рис. 8. Розпараллелені вертикальний та діагональний кроки

Після чотирнадцяти раундів перетворення функції G функція стиску виконує останній фінальний крок. Вона генерує нові 256-біт геш-значення, h^{i+1}, як великий XOR попереднього геш-значення h^i, значення *salt s* та 512-біт матриці стану *v* (рис. 9). Якщо кінцевий блок розширенного вхідного потоку був оброблений, тоді це являється результатом кінцевого виходу геш-функції BLAKE.

$$
\begin{aligned}
h'_0 &\leftarrow h_0 \oplus s_0 \oplus v_0 \oplus v_8 \\
h'_1 &\leftarrow h_1 \oplus s_1 \oplus v_1 \oplus v_9 \\
h'_2 &\leftarrow h_2 \oplus s_2 \oplus v_2 \oplus v_{10} \\
h'_3 &\leftarrow h_3 \oplus s_3 \oplus v_3 \oplus v_{11} \\
h'_4 &\leftarrow h_4 \oplus s_0 \oplus v_4 \oplus v_{12} \\
h'_5 &\leftarrow h_5 \oplus s_1 \oplus v_5 \oplus v_{13} \\
h'_6 &\leftarrow h_6 \oplus s_2 \oplus v_6 \oplus v_{14} \\
h'_7 &\leftarrow h_7 \oplus s_3 \oplus v_7 \oplus v_{15}
\end{aligned}
$$

Рис. 9. Фінальний крок стиску

1.4. Відмінності варіантів алгоритму

Другі три варіанти геш-функції дуже схожі на BLAKE-256. 512-бітна версія подвоює бітовий розмір більшості змінних: всі серединні геш-значення, матриця стиску, максимальний розмір повідомлення, сіль та значення лічильника. Вектор ініціалізації обраний з SHA-512, а константа c просто розширена більшою кількістю цифр з ϖ. Таблиця перестановок σ залишається без змін.

Єдиною нетрівіальною різниця в функції G: кількість bit-rotation визначені не просто подвоєнням попередньої кількості. Також в 512-бітній версії рекомендовано збідьшити кількість раундів з 14 до 16. Але фактично, цей параметр є настраюванним параметром користувача, як один з параметрів компромісу швидкості та захисту.

224- та 384-бітні версії використовують 256-бітну та 512-бітну версії алгоритму, потім урізають відповідно вихідне геш-значення до потрібного. Швидкодія цих реалізацій алгоритму ріна до тих що вже були розглянуті. Єдиною істотною різницею є вектори ініціалізації (використовуються SHA-224 та SHA-384 відповідно) та та виключення одиничних бітів («1»), під час розширення повідомлення до розміру вхідного блоку.

1.5. Надійність

Архітектура криптографічних геш-функцій передбачає балансування між рівнем захищеності

та швидкодією (а в аппаратних реалізаціях також затратами на електороенергію). В геш-функції BLAKE параметром, який дозволяє регулювати баланс захищеності та швидкодії виступає число раундів. З неофіційною метою значного підвищення рівня захисту відносно SHA-2, початкова кількість раундів SHA-2 була 10 раундів для 256-бітної версії та 14 раундів для 512-бітної версії. А вже в грудні 2010 кількість раундів була підвищена до 14 та 16 раундів відповідно, з метою забезпечення дуже високого рівня захищеності.

Під час розробки алгоритму, його автори приділили велику увагу питанню розсіювання даних, або лавинному ефекту. «Повне розсіювання» означає, що змінення одного біту вхідних даних (повідомлення, «солі» *salt* або вектору ініціалізації) може справити вплив на кожен біт виходу, причому досягається він за два раунди перетворень. Фунція стиску та функція G були спроектовані з метою мінімізації ймовірності локальних колізій, які з'являються коли два різних повідомлення призводять до однакового внутрішнього стану після деякого числа раундів. Крім того, значення *salt* (яке зазвичай тримається в секреті користувачем) передається до кожної функції стиску (а не тільки під час ініціалізації або фіналізації) для запобігання визначення значення *salt* із відомих пар повідомлення – геш-значення.

Розробники BLAKE також представили чотири зменшенні версії , тобто навмистно ослаблені версії для криптоаналітичних цілей.

Такими версіями виступають:

- BLOKE: всі *y* перетворення є проті від 0-15, по черзі
- FLAKE: функція стиску не містить *salt* або поточне геш-значення в її фінальному XOR-перетворенні
- BLAZE: функція G використовує нульові значення замість констант *c* отриманих з π
- BRAKE: дуже слабкий, містить всі три попередні слабкості

Методи криптоаналізу слабких версій перевіряються і на повній версії алгоритму. На грудень 2010 найкращою атакою, аку прийняли розробники геш-функції, була атака першого прообразу на 2.5 раундах, на вповній версії BLAKE із зменшеною кількістю раундів[2].

2. РЕАЛІЗАЦІЯ АЛГОРИТМУ

Так як внутрішня структура алгоритму BLAKE була успішно викладена та проаналізована, наступний розділ описує деякі методи практичної реалізації алгоритму. Цей розділ зфокусований на швидкодії, та не бере до уваги захищеність чи аппаратні вимоги.

2.1. ОФіційна реалізація BLAKE

Розробники функції гешування BLAKE запропонували декілька своїх реалізацій виконаних мовою програмування C. Разом із версіями простими для розуміння, які повністю відповідають опису алгоритму автори пропонують оптимізовані версії зі специфічними розмірами ключів та архітектурами вцілому. Офіційнийвеб-ресурс також містить посилання на одобрені та затверджені імплементації мовами Perl, PHP, Javascript та інші.

Коротко проаналізуємо спрощену 512-бітну версію (lightweight version). Так як компактна та призначена для одного варіанту вихідного розміру, вона вимагає мінімальних ресурсів програмного та аппаратного забеспечення та може працювати на багатьох платформах. Також в цю версію включено один Вбудований Тест (BIST), який гешує два вручну закодованих повідомлення та порівнює результат з вже відомими правильними геш-значеннями.

Геш-функція визивається як:

blake512_hash(digest, data, length), де

digest – це вказівник на 512-бітний вихідний буфер,

data – вказівник на вхідне повідомлення,

length – довжина вхідого повідомлення (у байтах).

Функція створює матрицю стану 4x4 та викликає три підпрограми : одна для ініціалізації станів, друга – для головного перетворення та третя для створення фінального стану повідомлення. Це відповідає основним крокам представлених в розділі з описом BLAKE-256.

Фунція blake512_update відповідає за розширення вхідного повідомлення до розміру оброблюваного блоку, розбиває повідомлення на 1024-бітні блоки, та ітеративно виконує функцію стиску. Функція стиску називається blake512_compress, вона має 16 раундів, кожний з яких містить вісім перетворень G, які використовують арифметичні макроси. Конкретно дана реалізація виконує всі вісім перетворень послідовно, вона не оптимізована для використання техніки розпаралелювання, яка була розглянута раніше.

Результат записується у digest та звичайно представляється, як 128 символьна строка у шістнадцятиричній системі. За замовчуванням, программа гешує повідомлення довжиною у 144 байти (1152 біта), але ця величина може бути змінена для тестування з різними довжинами. У код программи додається таймер для вимірювання швидкості гешування та фіксування результатів під час тестування.

Попередні результати тестування з використанням 64-бітного процесору Intel Core Duo представлені в табл. 1. Кількість тактів процесору на кожний вхідний байт для різних кількостей раундів.

Таблиця 1

Швидкодія / Розмір повідомлення

Назва геш-функції	Кількість раундів	Байти повідомленя			
		10	100	1000	10000
BLAKE-256	10	~1600	36.4	18.4	16.7
BLAKE-256	8	~1600	32.2	15.4	13.8
BLAKE-256	5	~1600	26.9	10.9	9.6
BLAKE-512	14	~1900	33.7	13.8	12.3
BLAKE-512	10	~1900	29.9	11.6	9.3
BLAKE-512	7	~1900	26.8	8.5	7.2

Результати ECRYPT тестування BLAKE-512 з використанням 64-бітного процесору Intel Core Duo представлені в табл. 2. Середні значення геш тактів на байт, для різних вхідних довжин.

Таблиця 2

Середні значення геш тактів на байт

Байти повідомлення	Q1	Median	Q2
8[1]	159.00	316.50	316.50
64[1]	19.69	38.81	39.00
576	9.12	9.19	9.19
1536	8.61	8.62	8.63
4096	8.02	8.03	8.05
довге[2]	7.65	7.69	7.74

[1]результати значно відрізняються
[2] приблизний асимптотичний ліміт

2.2. Комп'ютерна швидкодія

Специфікація BLAKE містить деякі тести швидкодії програмної реалізації на мікроконтроллерах та побутових (споживчих) процесорах CPU. Було відмічено, що 64-бітні версії значно швидчі за 32-бітні, кількість тактів на байт повідомлення зменшується, якщо росте розмір входу (так як розрахунок обислювальних витрат розподіляється на більшу кількість оброблених блоків), в той час як результати для маленьких довжин повідомлення (наприклад 10 байтів) є приблизними, з причини малого числа оброблених байтів які значно впливають на відношення такт/байт.

Відомий проект ECRYPT провів незалежне порівняння п'яти SHA-3 фіналістів на великій кількості комп'ютерних систем, та з різними довжинами повідомлень. Ці результати публічно представлені та доступні на веб-ресурсі проекту[3]. Таблиця 2 узагальнює ці результати для BLAKE-512 виконаному на 64-бітному процессорі Intel Core 2 Duo 2.4 GHz, для порівняння з результатами представленими у Таблиці 1. Результати також представлені на рис. 10, ілюструючи підвищення швидкодії (зменшення відношення тактів/байт) при збільшені розміру повідомлення.

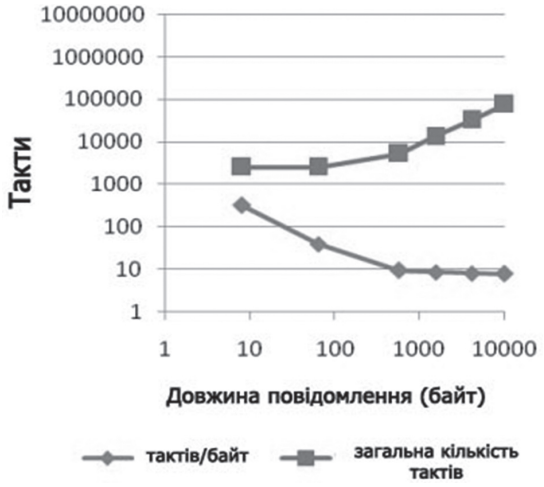


Рис. 10. Результати тестів ECRYPT для BLAKE-512 (Core 2 Duo)

Проект ECRYPT слугує як інформативне джерело порівняння геш-функцій фіналістів SHA-3: BLAKE, JH, Skein, Keccak та Grostl. За результатами даного проекту BLAKE є одним з найшвидших алгоритмів (можливо з причини невеликої кількості раундів), навіть швидший за Skein, та був обраний, як неофіційний переможець у власному конкурсі ECRYPT.

Кандидати SHA-3 також були проаналізовані на наявність «особливих» архітектур. Дослідники з Лабораторії Криптографічних алгоритмів протестували швидкодію всіх кандидатів другого раунду конкурсу SHA-3 використовуючи два спеціалізованих процессора: Cell Broadband Engine створений Sony, Toshiba та IBM, а також графічний процесор (GPU), розроблений NVIDIA. Обидва процесора є високоефективними, багатоядерними процесорами, які використовують модель Single Instruction Multiple Data (SIMD) та Single Instruction Multiple Threads (SIMT)[4]. GPU звичайно використовується з метою обробки важкого графічного контенту, в той час як Cell процессор використовується, наприклад, в самих передових PlayStation 3 ігрових приставках.

Ці тести використовують також різні типи метрики для вимірювання швидкодії. Кандидати були поділені на групи, AES натхненні та не AES натхненні (до яких і відноситься BLAKE). Тести останньої групи були сфокусовані на їх функціях стиску, більше ніж на весь алгоритм. Не враховуються час на налаштування системи, час фіналізації, або час копіювання даних, окрім копіювання ланцюга значень (таких як h^i для BLAKE). Ці досліди також сфокусовані на кількості разів, коли функція буде визвана, замість тактів. Такі дослідження важливі, тому що переможець SHA-3 буде реалізований на великій кількості сучасних спеціалізованих платформ.

ВИСНОВКИ

В даній статті була розглянута та проаналізована функція гешування BLAKE, один з фіналістів конкурсу на стандарт функції гешування SHA-3. Була розглянута історія створення, можливе застосування, архітектура та реалізація алгоритму. Його криптографічна структура та алгоритм були детально досліджені, зазначаючи на фактах які доводять захищеність та швидкодію. На практиці було реалізовано програмне забезпечення використовуючи за основу реалізацію представлену авторами функції, часткову реалізацію методів.

Майбутне алгоритму BLAKE та його потенційне світове впровадження буде відомо вже в поточному році, коли NIST оголосить, яка геш-функція стане новим стандартом SHA-3. Ці дослідження мають цінність для України, по-перше, тому що функція, яка війде до міжнародного стандарту функцій гешування буде впроваджена майже в усі сучасні інформаційні системи, по-друге, за досвідом конкурсу на стандарт симметричного шифру, навіть якщо BLAKE не буде

прийнятий, як стандарт, він буде реалізований в багатьох системах та пристроях.

Література

[1] J.-P. Aumasson, L. Henzen, W. Meier, and R. Phan, "SHA-3 proposal BLAKE," December 2010.

[2] L. Ji and X. Liangyu, "Attacks on round-reduced BLAKE," 2009.

[3] D. J. Bernstein and T. Lange, "List of SHA-3 candidates measured, indexed by machine," 2011, http://bench.cr.yp.to/results-sha3.html.

[4] J. W. Bos and D. Stefan, "Performance analysis of the SHA-3 candidates on exotic multi-core architectures," 2010.

[5] S. Neves, "ChaCha implementation," 2009, URL http://eden.dei.uc.pt/sneves/chacha/chacha.html.

[6] M. Knezevic, K. Kobayashi, J. Ikegami, S. Matsuo, A. Satoh, U. Kocabas, J. Fan, T. Katashita, T. Sugawara, K. Sakiyama, I. Verbauwhede, K. Ohta, N. Homma, and T. Aoki, "Fair and consistent hardware evaluation of fourteen round two SHA-3 candidates," April 2011.

Надійшла до редакції 13.03.2012

Кутя Євген Юрійович, магістрант кафедри БІТ ХНУРЕ. Область наукових інтересів: аналіз асиметричних криптосистем і функцій гешування, генератори ПВП, асиметричні криптопримітиви в групі точок еліптичних кривих.

Горбенко Іван Дмитрович, фото та відомості про автора див. на с. 190.

УДК 621.3.06

Анализ, сравнение и особенности архитектуры функции хеширования BLAKE проекта SHA-3 / Е.Ю. Кутя, И.Д. Горбенко // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 228–233.

Проводится анализ перспективной функции хеширования, претендента к принятию в качестве американского стандарта хеширования. Приводятся результаты оценки быстродействия алгоритма в зависимости от увеличения размера входного сообщения, а также возможности реализации параллельного вычисления значения хеш.

Ключевые слова: алгоритм хеширования, коллизия, быстродействие, стойкость, функция сжатия, начальное состояние, однонаправленность, итеративная структура.

Табл. 2. Ил. 10. Библиогр.: 6 наим.

UDC 621.3.06

Analysis, comparison and structure features of SHA-3 project's hash function BLAKE / E.Yu. Kutia, I.D. Gorbenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 228–233.

The analysis of a perspective hash function is made which is an applicant for the acceptance as the American hash function standard. Algorithm performance evaluation results depending on input message size and possibilities of implementing parallel hash value computing are presented in this paper.

Keywords: hashing algorithm, collision, performance, attack resistance, compress function, initial state, inconvertibility, iterative structure.

Tab. 2. Fig. 5. Ref.: 5 items.

УДК 512.624.95 + 517.772

МЕТОД ГЕНЕРАЦИИ ПСЕВДОСЛУЧАЙНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ НА ОСНОВЕ ИЗОМОРФНЫХ ТРАНСФОРМАЦИЙ ЭЛЛИПТИЧЕСКОЙ КРИВОЙ

А.В. БЕССАЛОВ, В.Е. ЧЕВАРДИН

Метод генерации случайных битовых последовательностей, основанный на изоморфных трансформациях точек эллиптической кривой. Предложенный метод отличается от существующих увеличением числа внутренних состояний. Это позволило повысить сложность восстановления закона формирования случайных последовательностей. Одним из результатов является получение нижней границы периода случайной последовательности.

Ключевые слова: генератор псевдослучайных последовательностей, эллиптические кривые, изоморфные трансформации, трансформации эллиптической кривой.

ВВЕДЕНИЕ

Одним из важнейших направлений в современной криптографии является разработка и усовершенствование алгоритмов генерации псевдослучайных последовательностей (ПСП). Широкое признание получили алгоритмы генерации ПСП: IEEE 182.3, DRBG block cypher (DRBGBC) – генераторы на основе блочных шифров [9], BBS и другие. Достоинством этих генераторов ПСП является достаточно высокая скорость формирования ПСП. Их криптографическая стойкость считается эквивалентной стойкости примитива, используемого в качестве раундовой функции. Так, к примеру, стойкость генератора IEEE 182.3 эквивалентна стойкости криптографического примитива, лежащего в основе блочно-симметричного шифра DES, стойкость DRBGBC эквивалентна стойкости AES [9], стойкость генератора BBS основана на сложности решения задачи факторизации целого числа, стойкость генератора Dual_EC_DRBG [9] эквивалентна сложности решения задачи дискретного логарифмирования в группе точек кривой.

Учитывая большее доверие к криптопреобразованиям с теоретически доказуемой стойкостью, основное внимание современных исследований устремлено на разработку теоретически стойких криптографических методов на основе преобразований в группе точек эллиптической кривой (ЭК) [1-7]. Ярким примером являются алгоритмы генерации ПСП на ЭК. Однако, существующие алгоритмы построения генераторов ПСП на ЭК [1-7] отличаются высокой вычислительной сложностью, что существенно ограничивает их область применения и конкурентоспособность алгоритму BBS.

В связи с этим, целью данной работы является разработка нового метода генерации ПСП на основе арифметики ЭК, что позволит увеличить число внутренних состояний генератора за счет использования множества изоморфизмов базовой ЭК и, как следствие, сложность восстановления закона формирования ПСП. Это в свою очередь позволит уменьшить характеристику поля Галуа и снизить вычислительные затраты при генерации ПСП без снижения криптографической стойкости генератора ПСП.

1. ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

Пусть гладкая ЭК над простым полем Галуа характеристики $p \neq 2,3$ [8], $E[F_p]$ задана уравнением в канонической форме:

$$EC: y^2 = x^3 + a_4 x + a_6 \pmod p, \qquad (1)$$

где $a_4, a_6 \in F_p$.

Точки кривой представлены двумя координатами $\{X, Y\} \in F_p$, удовлетворяющими уравнению (1), $P_i = (X_{P_i}, Y_{P_i}) \in E_p$, где E_p – абелева группа точек ЭК. Базовой операцией является скалярное произведение точки[1]. Сложность вскрытия криптосистем, основанных на решении задачи дискретного логарифмирования в группе точек EC[2], определяется ρ-методом Полларда $\approx O(\sqrt{n})$ операций сложения точки кривой.

Для ЭК в форме (1) существует изоморфная трансформация $\varphi: \{u, r, s, t\}$ [8]:

$$\varphi(u,r,s,t) = \begin{cases} X = u^2 X' + r, \\ Y = u^3 Y + s u^2 X' + t, \end{cases} \qquad (2)$$

где переменные $u, r, s, t \in F_p, u \neq 0$ пробегают все значения: 0..p-1.

Используя для базовой кривой EC фиксированный изоморфизм $\varphi(u,r,s,t)$, получим изоморфную кривую EC'. В таком случае, можем любую точку кривой EC однозначно трансформировать в точку изоморфной кривой EC'. Наличие изоморфной кривой дает возможность получить эквивалентную группу точек кривых, которая не является автоморфизмом базовой группы. Это означает, что последовательности точек изоморфных групп эквиваленты, но отличаются друг от друга. Представим изоморфные трансформации группы точек базовой кривой в виде матрицы (таблица).

[1] Скалярное произведение точки кривой – является сложением точки P с собой k раз, $kP = \underbrace{P + P + .. + P}_{k\ раз} \pmod p$, где $k < \#P$, $\#P$ – порядок точки P.

[2] Задача дискретного логарифмирования в группе точек кривой E – по известным параметрам Q, P, p, связанных выражением $P = kQ = \underbrace{Q + Q + .. + Q}_{k\ раз} \pmod p$, необходимо определить неизвестное k.

Таблица

Точки *EC* / Из. транс.	Q_1	Q_2	Q_3	...	Q_n
φ_1	$P^1{}_1$	$P^2{}_1$	$P^3{}_1$	...	$P^n{}_1$
φ_2	$P^1{}_2$	$P^2{}_2$	$P^3{}_2$	...	$P^n{}_2$
...	...	...	...	...	...
φ_{Nec}	$P^1{}_{Nec}$	$P^2{}_{Nec}$	$P^3{}_{Nec}$	...	$P^n{}_{Nec}$

Следовательно, количество различных последовательностей точек будет расти пропорционально числу трансформаций ЭК N_{EC}, что даст положительный эффект при построении генераторов ПСП. В существующих методах [2,3,5-7] для генерации ПСП используются точки из одной группы, соответствующей φ_1 из таблицы, кроме метода [2], в котором предлагается использовать две изоморфные кривые для построения однонаправленной функции. Как показали оценки мощности множества трансформаций ЭК, для канонической формы она растет пропорционально характеристике p поля Галуа, а для трансформации в нормальную форму рост происходит пропорционально p^4. Это свойство ЭК планируется использовать для увеличения числа внутренних состояний генератора ПСП на ЭК, что позволит увеличить нижнюю границу числа выходов генераторов этого класса.

2. СУЩЕСТВУЮЩИЕ МЕТОДЫ ГЕНЕРАЦИИ ПСП НА ОСНОВЕ МЕХАНИЗМА DRBG

В источниках [1-7] представлен ряд подходов к построению генераторов ПСП на основе сложения точек кривой, скалярного произведения, скалярного произведения на двух ЭК [2, 3], спаривания точек кривой [7]. Однако принятым в качестве стандарта является генератор Dual_EC_DRBG [9]. В нем задача восстановления закона формирования ПСП сводится к решению задачи дискретного логарифмирования в группе точек ЭК. Структура генератора DRBG представлена следующей моделью (рис. 1).

Как известно, одним из значимых компонентов такого механизма является источник энтропии, определяемый реализацией DRBG. Функция преобразования метки seed (Reseed) обеспечивает секретность выхода DRBG, если seed или внутреннее состояние стало известным.

Энтропия источника влияет на количество внутренних состояний (рис. 1), следовательно, и на нижнюю границу множества выходных состояний DRBG. Увеличение числа внутренних состояний позволит увеличить сложность восстановления закона формирования ПСП злоумышленником. Рассмотрим функциональную модель генератора Dual_EC_DRBG [9] (рис. 2).

Внутреннее состояние генератора определяется параметрами: (*s*, *seedlen*, *p*, *a*, *b*, *n, P, Q, security_strength, prediction_resistance_flag, reseed_counter*) [9], где *P, Q* – базовые точки кривой порядка *n*, *s* – секретный скаляр.

Функция генерации представлена выражением (3).

$$r_i = \phi(X\left[\phi(X\left[t_{i-1} * P\right]) * Q\right]), \qquad (3)$$

где s – секретное число; $t_0 = seed = hash(s)$; r_i – выход генератора.

Число выходов генератора Dual_EC_DRBG равно числу координат точек циклической

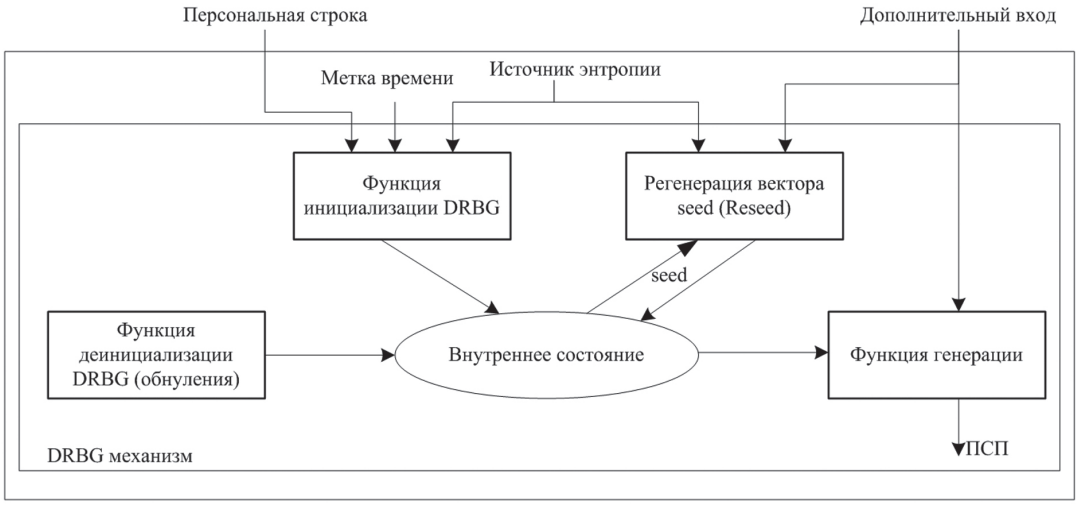


Рис. 1. Функциональная модель DRBG

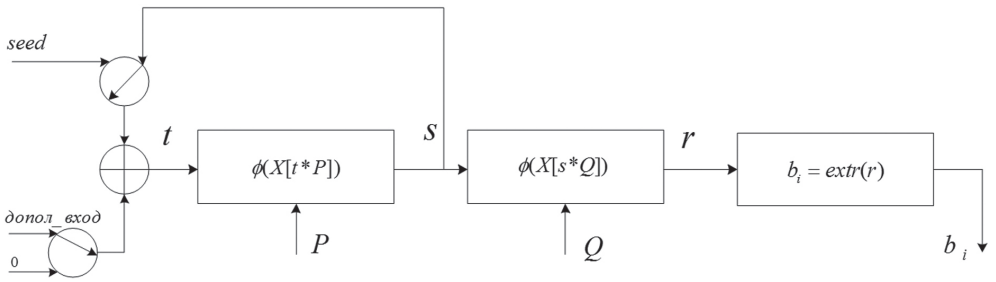


Рис. 2. Модель генератора Dual_EC_DRBG

группы точки Q, т.е. ограничено сверху порядком базовой точки, $n = \#Q$.

Нижняя граница периода Dual_EC_DRBG определяется числом внутренних состояний. Для обеспечения требуемой стойкости эта граница должна быть не ниже n. Учитывая, что внутреннее состояние генератора определяется значением $X[t*P]$, число различных состояний будет не более $n/2$. Верхняя граница будет зависеть от способа генерации скаляра t. Таким образом, возникает ситуация, когда нижняя граница периода ПСП будет равна числу различных $X[t*P]$.

Рассмотрим одну из возможностей, позволяющих увеличить число внутренних состояний генератора Dual_EC_DRBG.

3. МЕТОД ГЕНЕРАЦИИ ПСП НА ОСНОВЕ ИЗОМОРФНЫХ ТРАНСФОРМАЦИЙ ТОЧЕК ЭК

Пусть задана базовая ЭК в канонической форме, EC. Изоморфные трансформации этой кривой заданы выражением (2). Для описания алгоритма генерации генератора (рис. 3) зафиксируем следующие структурные элементы:

1. Базовая эллиптическая кривая EC.
2. Базовые точки кривой – P и Q.
3. Операция получения изоморфной базовой точки $P_i = \varphi_i(P)$.
4. Операция получения текущей точки кривой: $f(P_{i-1}, P_i) = P' = t_i * P_i$.
5. Операция извлечения битов из координаты X текущей точки кривой: $r_i = \phi(X[P_i])$ согласно [9].

Представим функцию генерации текущей точки P':

$$f(P_{i-1}, \varphi_j(P)) = P' = t_i * \varphi_i(P). \quad (4)$$

Используя выражение (4) представим функцию генерации:

$$r_i = \phi(X\left[\phi(X\left[P_i\right])*Q\right]) = \phi(X\left[(\phi(X\left[t_i*\varphi_i(P)\right])*Q\right]), \quad (5)$$

где ϕ – функция преобразования координаты X в целое число.

Изначально устанавливается состояние генератора: вводится характеристика p поля Галуа, коэффициенты базовой кривой EC, базовые точки P и Q, требуемая длина ПСП $l_{\text{ПСП}}$ ($l_{\text{ПСП}}$ задает количество итераций). С помощью однократного преобразования базовой точки кривой EC (скалярного умножения) получаем на каждой итерации новую точку $P' = t_i * \varphi_i(P)$. Последовательность таких точек кривой будет обладать периодом, равным порядку циклической группы точек n. Кроме операции над базовой точкой в своей группе будем каждую итерацию трансформировать точку базовой кривой в изоморфную, $P_i = \varphi_i(P)$ (2). Результат после второго скалярного произведения $b_i = extr(\phi(X(s_i * Q)))$ будет элементом ПСП (рис. 3). Результат произведения $P' = t_i * P_i$ пробегает все точки таблицы.

С целью повышения сложности восстановления внутренних состояний DRBG изоморфизм можно выбирать специальной функцией, задающей параметры изоморфизма $\varphi_i = \{u_i, r_i, s_i, t_i\}$ определенным образом (по случайному закону или в определенном порядке). Далее рассмотрим один из вариантов функции, генерирующей значения u_i изоморфизма $\varphi_i = \{u_i, r_i, s_i, t_i\}$.

Для получения текущей базовой точки P_i, зафиксируем генератор ω группы Z_p, где p – характеристика поля Галуа. Затем, учитывая, что u пробегает все значения вычетов в поле p, текущее значение u_i получим:

$$u_i = \omega^{2i} \bmod p = u_{i-1} * \omega^2 \bmod p. \quad (6)$$

Число изоморфных точек базовой точки равно $N_{EC} = \frac{1}{2}(p-1)$. Параметр u пробегает все значения $\{1,..,N_{EC}\}$.

Для получения текущего значения скаляра t, будем использовать генератор ω' группы Z_n, где n – порядок циклической группы точек кривой (простое число), которой принадлежат точки P и Q.

Текущее значение скаляра t_i получим следующим образом:

$$t_i = t_{i-1} * t \bmod n, \quad (7)$$

где t – генератор мультипликативной группы Z_n.

Для обеспечения криптографической стойкости генератора (рис. 3) значение ω' будем использовать в преобразованном виде. Для этого выбирается секретное число $seed$, так что $(seed, n) = 1$. Число t определяется выражением (8).

$$t = \omega'^{seed} \bmod n, \quad (8)$$

где ω' – генератор Z_n.

Учитывая выражение (8) выражение для t_i примет вид:

$$t_i = t_{i-1} * t \bmod n = t_{i-1} * \omega'^{seed} \bmod n \quad (9)$$

Очевидно, что t_i пробегает все значения группы Z_n.

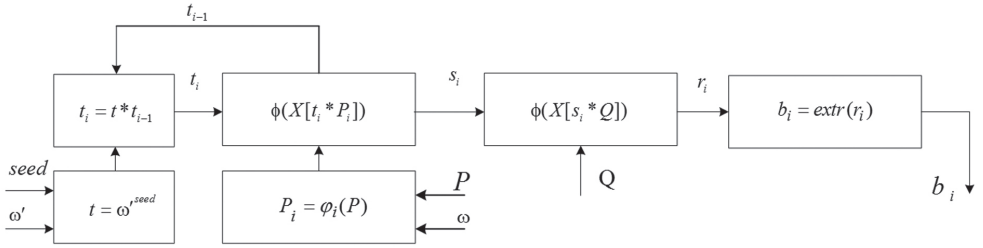


Рис. 3. Модель модифицированного генератора Dual_EC_DRBG

Учитывая граничные значения числа изоморфизмов ЭК в канонической форме, $N_{EC} = \frac{1}{2}(p-1)$, число внутренних состояний модифицированного генератора Dual_EC_DRBG определяется значением (10).

$$N = \frac{1}{2}(p-1) * n, \qquad (10)$$

где n – порядок циклической группы точек кривой; p – характеристика поля Галуа.

ВЫВОДЫ

Таким образом, разработан новый метод генерации ПСП на основе применения изоморфных трансформаций точек ЭК. Получено аналитическое выражение (10) для оценки числа внутренних состояний генератора Dual_EC_DRBG с использованием предложенного метода. Полученный метод позволяет в $\frac{1}{2}(p-1)$ раз увеличить число внутренних состояний генератора Dual_EC_DRBG, что увеличивает сложность вскрытия закона формирования ПСП злоумышленником. Применение разработанного метода также позволяет избежать существующих недостатков Dual_EC_DRBG.

Для обеспечения более высокой криптографической стойкости полученного метода генерации ПСП значение u_i можно получать аналогичным образом, на основе секретного числа *seed*.

При фиксированном значении числа внутренних состояний генератора Dual_EC_DRBG разработанный метод позволит сократить битовую длину характеристики p поля при фиксированной стойкости генератора. Следует также отметить применимость полученного метода ко всем генераторам ПСП на ЭК.

Литература

[1] Kaliski Jr. B. S. A pseudo-random bit generator based on elliptic logarithms / B. S. Kaliski Jr. // Advances in Cryptology: Proceedings of Crypto '86 (Lecture Notes in Computer Science, vol. 263), Springer-Verlag, New York, 1987, pp. 84-103.

[2] Impagliazzo R. Pseudo-random generation from one-way functions / R. Impagliazzo, L. Levin, and M. Luby // Proceedings of the 21st Annual ACM Symposium on Theory of Computing, ACM, New York, 1989, pp. 12-24.

[3] *Burton S.* One-Way Permutations on Elliptic Curves / Burton S. Kaliski, Jr. // Journal of Cryptology (1991) International Association for Cryptologic Research. 1991. - P.187-199.

[4] *Гриненко Т.А.* Методы формирования псевдослучайных последовательностей в группах точек эллиптических кривых / Т.А. Гриненко, С.И. Збитнев, Д.В. Мялковский // Радиотехника: Всеукр. межвед. науч.-техн. сб. 2002. Вып.119.С.119-123.

[5] *Gjusteen K.* Comments on Dual-EC-DRBG/NIST SP 800-90, Draft December 2005 / Kristian Gjusteen // March 16, 2006.

[6] *Гріненко Т.О.* Дослідження властивостей генераторів псевдовипадкових бітів на еліптичній кривій на відповідність міжнародному стандарту ISO/IEC 18031 / Гріненко Т.О., Погребняк К.А. // Журнал "Прикладная радиоэлектроника" 2009 №3. Харьков - 2009, сс. 372 – 377.

[7] *Горбенко І.Д.* Метод побудування випадкових бітів на основі спарювання точок еліптичних кривих / Горбенко І.Д., Шапочка Н.В., Погребняк К.А. // Журнал "Прикладная радиоэлектроника" 2010 №3. Харьков - 2010, сс. 386 - 394.

[8] *Husemцller D.* Elliptic Curves, Second Edition // Springer – 2002 / Dale Husemцller; with appendices by Stefan Theisen, Otto Forster, and Ruth Lawrence. - p. cm. — (Graduate texts in mathematics; 111) Includes bibliographical references and index. ISBN 0-387-95490-2 (alk. paper).

[9] NIST Special Publication 800-90 Recommendation for Random Number Generation Using Deterministic Random Bit Generators (Revised) / Elaine Barker, John Kelsey // Computer Security Division Information Technology Laboratory National Institute of Standards and Technology. - March 2007.

Поступила в редколлегию 22.03.2012

Бессалов Анатолий Владимирович, фото и сведения об авторе см. на с. 226.

Чевардин Владислав Евгеньевич, кандидат технических наук, докторант ВИТИ НТУУ «КПИ». Область научных интересов: криптографическая защита информации.

УДК 512.624.95 + 517.772

Метод генерації псевдовипадкових послідовностей на основі ізоморфних трансформацій еліптичної кривої / А.В. Бессалов, В.Є. Чевардін // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 234–237.

Запропоновано метод генерації випадкових бітових послідовностей оснований на ізоморфних трансформаціях точок еліптичної кривої. Метод відрізняється від існуючих підвищенням числа внутрішніх станів. Це дозволило підвищити складність відтворення закону формування випадкової послідовності. Одним з результатів також є отримання нижньої границі періоду випадкової послідовності.

Ключові слова: генератор псевдовипадкових послідовностей, еліптична крива, ізоморфні трансформації, трансформації еліптичної кривої.

Табл. 01. Іл. 03. Бібліогр.: 9 найм.

UDC 512.624.95 + 517.772

A method of generating pseudorandom sequences based on elliptic curve isomorphic transformations / A.V. Bessalov, V.E. Chevardin // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 234–237.

A method of generating random bit sequences based on isomorphic transformations of elliptic curve points is proposed. The method is different from the existing ones by an increased number of internal states. It has allowed to increase the complexity of restoring the law of forming random sequences. One of the results is also obtaining the lower boundary of a random sequence period.

Keywords: pseudorandom sequence generator, elliptic curves, isomorphic transformations, elliptic curve transformations.

Tab. 01. Fig. 03. Ref.: 9 items.

УДК 681.3.06

О НЕКОРРЕКТНОСТИ СТАНДАРТНОГО УСЛОВИЯ ДЛЯ MOV-АТАКИ НА ЭЛЛИПТИЧЕСКИЕ КРИВЫЕ

А.В. БЕССАЛОВ

Дан анализ двух условий для бидинейного спаривания точек и MOV- атаки, определенных теоремой Кассельса. Замечено, что стандартное условие для MOV- атаки отвечает лишь одному из условий теоремы Кассельса и не гарантирует возникновение нециклической структуры группы точек. Предложено в будущих стандартах добавить к тесту проверку наличия квадрата в порядке кривой.

Ключевые слова: эллиптические кривые, кривые Эдвардса, порядок кривой, порядок точки, простое поле, расширенное поле.

ВВЕДЕНИЕ

Широко известная в сфере криптографии работа [1] предложила одну из первых атак изоморфизма на проблему дискретного логарифмирования (DLP) в группе точек эллиптической кривой. Эта атака, получившая по именам авторов название MOV-атаки, сводится к отображению пары точек кривой E над некоторым расширением F_q^k поля F_q в элемент поля расширения, что при небольших значениях k катастрофически снижает сложность DLP. Из большинства криптоприложений после этой работы были исключены уязвимые к MOV-атаке суперсингулярные кривые, а все появившиеся через десятилетие стандарты (в частности, [2–7]) включили обязательные тесты на стойкость кривой к MOV-атаке. Автор данной статьи обнаружил, что результат тестирования и действительная уязвимость кривой к MOV-атаке могут оказаться далекими друг от друга, в итоге отбраковываются достаточно стойкие кривые, но не выдержавшие тест. В статье для убедительности приводится простой пример, иллюстрирующий вышесказанное.

УСЛОВИЯ ДЛЯ MOV-АТАКИ И УСЛОВИЯ ТЕСТИРОВАНИЯ В СТАНДАРТАХ

Пусть $N_E = hn$ – порядок кривой E над конечным полем F_q, где n – большое простое число, а кофактор h невелик (обычно $h \le 4$). Криптосистема строится на циклической подгруппе точек кривой простого порядка n.

Изоморфное отображение, рассмотренное в [1], строится как билинейное спаривание Вейля (или Тейта и др.) двух точек кривой в расширении F_q^k, $k = 1, 2, 3,...$, в котором возникает нециклическая группа $nG \times nG$ точек порядка n, содержащая n^2 точек. Спаривание необходимым образом использует две точки из разных циклических подгрупп порядка n нециклической группы порядка n^2 [9]. Можно, таким образом, утверждать, что достаточным условием для MOV-атаки является возникновение нециклической группы точек порядка n в некотором расширении поля F_q. В принципе нециклическая группа точек может существовать и в поле F_q ($k = 1$), если $h = cn$, но это не отвечает принятым ограничениям. Для суперсингулярных кривых нециклическая группа образуется уже при $k = 2..6$, для несуперсингулярных – значения k, как правило, достаточно велики и могут оказаться соизмеримыми с порядком поля q. В исключительных случаях группа $nG \times nG$ может возникнуть и при небольших значений k, что и требует MOV-тестирования всех, в том числе и несуперсингулярных кривых.

Необходимые условия для нециклической структуры группы E_q формулируются в теореме Кассельса [8, с.85]: группа E_q порядка $N_E = n_1 n_2$ является либо циклической, либо представляется прямой суммой двух циклических подгрупп порядков n_1 и n_2, таких что

$$n_1 \mid n_2 \text{ и } n_1 \mid \text{НОД}(N_E, q-1) \qquad (1)$$

Отсюда, в частности, следует, что порядок N_E содержит квадрат n_1^2. Заметим, что выполнение обоих условий (1) еще не гарантирует нециклической структуры группы (хотя, как правило, это так). Многие циклические кривые, например, содержат квадраты в порядке кривой [8]. Необходимые и достаточные условия нециклической структуры эллиптической кривой для общего случая пока не определены.

Обратимся теперь к известным стандартам [2-7]. Тест на стойкость кривых к MOV-атаке в них состоит в проверке неделимости $n \nmid q^k - 1$ для всех $k = 1..B$, с возможно различными значениями верхней границы B. Этот тест, отвечающий лишь второму условию (1) теоремы Кассельса, даже не гарантирует квадрата n^2 в порядке кривой в расширении F_q^k. Его можно было бы классифицировать как «подозрение на уязвимость к MOV-атаке». Конечно, это не снижает безопасности проектируемых криптосистем, однако вряд ли обоснованным (и корректным) является упрощенный тест, отвергающий приемлемые для криптосистем кривые. По-видимому, более целесообразно при доработке стандартов усилить этот тест по меньшей мере дополнительной проверкой на наличие квадрата n^2 в порядке кривой над большим полем.

Пример. Для иллюстрации примем порядок точки $n = 3$ и построим несуперсингулярную кривую

$$y^2 + xy = x^3 + ax^2 + b, \; b \ne 0 \qquad (2)$$

над полем F_2^2 ($q = 2^2$) с неприводимым полиномом $P(x) = x^2 + x + 1$ и его корнем α, для которого $\alpha^2 + \alpha + 1 = 0$. Здесь α – генератор мультипликативной

группы поля F_2^2 3-го порядка ($\alpha^3 = 1$) со следом 1. В границах Хассе 2..8 с четными значениями N_E нас устраивает лишь кривая с порядком $N_E = 6 \equiv 2 mod 4$, поэтому коэффициент a в (2) должен иметь след 1[8]. Примем $a = \alpha$. Для точки $Q = (x_1, y_1)$ 3-го порядка $2Q = -Q$ нетрудно получить уравнение

$$x_1^4 + x_1^3 + b = 0, \qquad (3)$$

которое в нашем случае ($x_1^3 = 1$) принимает вид $x_1 = 1 + b$. Значения $b \neq 0, 1$ (в последнем случае получим точку 2-го порядка), поэтому примем $b = \alpha$, тогда кривая $y^2 + xy = x^3 + \alpha x^2 + \alpha$ имеет точки 3-го порядка $Q = (\alpha^2, \alpha^2)$, $-Q = (\alpha^2, 0)$, а порядок кривой $N_E = 6$. Так как $N_E = q + 1 - t_1$, то параметр t_1 (след уравнения Фробениуса) равен $t_1 = -1$.

Найдем порядки этой кривой над над расширениями $F_q^2 = F_2^4$ и $F_q^3 = F_2^6$. Рекуррентная формула расчета параметра t_k имеет вид [8]

$$t_{k+2} = t_1 t_{k+1} - q t_k, \quad k = 0, 1, 2, \ldots, \ t_0 = 2.$$

Отсюда $t_2 = -7$, $N_{E2} = q^2 + 1 - t_2 = 24$, $t_3 = 11$, $N_{E3} = q^3 + 1 - t_3 = 54$. Хотя в этом примере $n|(q-1)$ и $n|(q^2-1)$, о билинейном спаривании (или MOV-атаке) речи не идет, поскольку порядки соответствующих кривых 6 и 24 не делятся на 3^2. И только расширение степени $k = 3$ с выполнением $n|(q^3 - 1)$ дает нециклическую группу типа (2, 3, 3^2), имеющую ровно 3^2 точек порядка 3. Порядок кривой $N_{E3} = 54 = 2 \cdot 3^3$ содержит квадрат 3^2 и выполняются оба условия теоремы Кассельса (1). Кроме того, кривая действительно содержит 9 точек 3-го порядка (что и является условием нециклической группы точек 3-го порядка). Действительно, уравнение (3)

$$x_1^4 + x_1^3 + \beta^{21} = 0, \qquad (4)$$

в поле F_2^6 с примитивным элементом β, для которого $\beta^6 + \beta + 1 = 0$, имеет ровно 4 решения. В уравнении (4) $\beta^{21} = \alpha$ – элемент подполя F_2^2 3-го порядка. Одно из решений (4) лежит в этом подполе и является тривиальным $x_1^{(1)} = \alpha^2 = \beta^{42}$. Остальные 3 решения принадлежат расширению F_2^6 и равны

$$x_1^{(2)} = \beta^{23} = 101001, \quad x_1^{(3)} = \beta^{29} = 111000,$$
$$x_1^{(4)} = \beta^{53} = 101010.$$

Каждое из решений (4) дает по 2 точки 3-го порядка, в итоге вместе с точкой на бесконечности имеем 9 точек 3-го порядка и, следовательно, нециклическую структуру группы.

Итак, хотя тест на MOV-атаку в примере отбраковывает все кривые с расширениями степени $k = 1, 2, 3$, уязвимой к MOV-атаке является лишь кривая над полем F_q^3.

В заключение еще раз подчеркнем, что стандартный тест на MOV-атаку стал бы более корректным, если его усилить дополнительной проверкой: $n^2 \nmid N_{Ek}$, где N_{Ek} – порядок кривой E в расширении F_q^k.

Литература

[1] *Menezes A.J, Okamoto T., Vanstone S. A.* Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field. University of Waterloo, sep. 1990. And //IEEE Transactions on Information Theory, V39, 1993. – PP 1639-1646.

[2] *IEEE P1363-2000.* Standard Specifications for Public Key Cryptography. Institute of Electrical and Electronics Engineers, Inc.,2000.

[3] *ISO/IEC JTC* 1/SC 27 n 2303, CD 15946-2. Information Technology- Sequrity Techniques – Cryptographic Techniques based on Elliptic Curves: Part 2- Digital Signatures. 1999 -05-26..

[4] *ANSI X9.62-1999.* Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA). 1999.

[5] *FIPS 186-2.* Digital Signature Standard. National Institute of Standard and Technology. 2000.

[6] *ГОСТ Р 34.10-2001.* Информационная технология. Криптографическая защита информации. Процедуры формирования и проверки электронной цифровой подписи. – М.: Госстандарт России, 2001. – 20с.

[7] *Державний стандарт України ДСТУ 4145-2002.* Інформаційні технології. Криптографічний захист інформації. Цифровий підпис, що грунтується на еліптичних кривих. Формування та перевіряння. Київ, Держстандарт України, 2003. – 94с.

[8] *Бессалов А.В., Телиженко А.Б.* Криптосистемы на эллиптических кривых: Учеб. Пособие. – К.: ІВЦ «Видавництво «Політехніка»», 2004, 224 с.

[9] *Markus Jakobsson and Wenbo Mao.* Cryptographic Protocols. Prentice-Hall, 2006.

Поступила в редколлегию 5.04.2012

Бессалов Анатолий Владимирович, фото и сведения об авторе см. на с. 226.

УДК 681.3.06

Про некоректність стандартної умови для MOV-атаки на еліптичні криві / А.В. Бессалов // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 238–239.

Дано аналіз двох умов для білінійного спарювання точок і MOV- атаки, які визначені теоремою Касельса. Відмечено, що стандартна умова для MOV-атаки відповідає лише одній із умов теореми Касельса і не гарантує появу нециклічної структури групи точок. Запропоновано у майбутніх стандартах додати до тесту перевірку існування квадрату у порядку кривої.

Ключеві слова: еліптичні криві, криві Едвардса, порядок кривої, порядок точки, просте поле, розширене поле.

Бібліогр.: 9 найм.

UDC 681.3.06

On the incorrectness of a standard condition for MOV-attack to elliptic curves / A.V. Bessalov // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 238–239.

The analysis of two conditions of the Kassels theorem for bilinear pairing of points and for a MOV - attack is given. It is noticed that the standard condition for the MOV-attack corresponds only to one of the conditions of the Kassels theorem and does not guarantee the origin of a noncyclic structure of a group of points. It is offered to add to the MOV-test a quadrate presence one in a curve-order in the future standards.

Keywords: elliptic curves, Edwards curves, curve order, point order, prime field, extension field.

Ref.: 9 items.

УДК 681.3.07

СХЕМИ ЕЦП ДЛЯ ГРУП ПІДПИСІВ СКОРОЧЕНИХ ПОВІДОМЛЕНЬ

О.А. ШЕВЧУК

Запропоновано метод формування ЕЦП для множин невеликих повідомлень з відсутньою збитковістю. Запропоновано критерій ефективності подібних схем. Розглянуто ЕЦП з відновленням повідомлення в стандарту ISO/IEC 9796-3.

Ключові слова: ЕЦП, відновлення повідомлення, оптимізація обчислень.

ВСТУП

В сучасних інформаційних системах актуальні питання перевірки дійсності та справжності множин невеликих повідомлень, та подальше їх зберігання.

Наведемо модель, що розглядається. Нехай A передає B деякі повідомлення $m_1, m_2, \ldots m_\infty$. В деякий проміжок часу t_i A може передати $n \in [1;n]$ повідомлень як $M_i = \{m_{i,1}, m_{i,2}, \ldots m_{i,n}\}$. Абонентові B у будь-який проміжок часу $t_j, j > i$ необхідно встановити такі $m \in M_i, M_i \in \{M\}$, що не є цілісними та справжніми. В статті розглядається випадок для стислих m.

Пропонуються критерії порівняння схем ЕЦП, використовуючи які можливо зменшити постійні затрати на зберігання множин стислих повідомлень з їх цифровими підписами.

Задача є актуальною для систем перевірки цілісності компонентів програмного забезпечення, журналювання, мережевих протоколів тощо. Загальна схема

Запропонуємо наступне визначення стислого повідомлення. Нехай повідомлення m є стислим, якщо для забезпечення визначеного рівня стійкості до атаки селективної підробки необхідно геш значення h таке що $L_b(h) > L_b(m)$, де $L_b(x) = \lceil log_2 x \rceil$.

Оберемо показник ефективності схеми, як відношення доданої частки $\{H, S\}$ повідомлення до основної $\{M\}$:

$$\Delta = \frac{L_b(H) + L_b(S)}{L_b(M)}, \qquad (1)$$

де $M = \{m_1, m_2, \ldots m_n\}$ – повідомлення, такі що $\forall m \in M : L_b(m) = const$; $H = \{h_1, h_2, \ldots h_n\}$ – геш значення повідомлень та S – цифровий підпис H.

Ідеальною схемою будемо вважати таку, що задовольняє

$$\Delta \le 1 \qquad (2)$$

Загальний обсяг повідомлення складає

$$L_b(\{M, H, S\}) = \\ = L_b(S) + \sum_{\forall m \in M} (L_b(m) + L_b(Hash(m))) \qquad (3)$$

Якщо $\forall m : L_b(Hash(m)) = const$, доданий обсяг для $\{m_1, m_2, \ldots m_n\}$ складе

$$L_d = const = n \cdot L_b(Hash(0)) + L_b(S). \qquad (4)$$

Легко побачити, що не (2) задовольняється, коли $\forall m \in M : m \le L_b(Hash(m))$.

Задачею, що вирішується в статті, є пошук такої схеми цифрового підпису, що має найбільш подібну до (2) схему цифрового підпису.

1. ІСНУВАННЯ ІДЕАЛЬНОЇ СХЕМИ /ДЛЯ СКОРОЧЕНИХ ПОВІДОМЛЕНЬ

Для обчислення вірних параметрів значення $L_b(Hash())$ розглянемо складність атаки селективної та екзистенційної підробки для підписів з відновленням повідомлення у випадку використання схеми з множиною скорочених повідомлень.

Нехай зловмисник Z хоче створити підпис (r,s), $r \in [1, n-1]$ для деякого повідомлення m. Зловмисник має створити такі (r,s), що $V((r,s)) \neq \perp$. Для всіх схем ЕЦП обидві компоненти r,s обчислюються у групі точок ЕК за допомогою сеансового ключа k. Відновлення k має експоненційну складність. Тому Z має вгадувати r.

Встановимо ймовірність того, що Z вгадає геш значення обраного повідомлення. Нехай Z вгадує $h \in [1, 2^{L_{bh}}]$ для повідомлення $m \in [1, 2^{L_{bm}}]$, де

$$\begin{cases} \forall h \in H : L_{bh} = L_b(h) = const \\ \forall m \in M : L_{bm} = L_b(m) = const \end{cases}$$

Нехай Z за допомогою випадкової функції обирає деяке $r' \in [1, n-1], r' \neq r$. Зробимо припущення, що

$$\forall r' \in [1, n-1] : \exists H : Sign(H) = (r', s)$$

Тоді, якщо $H = \{h_1, h_2, \ldots h_k\}$, ймовірність вгадування геш значення деякого повідомлення $j \in [1, k]$ в множині геш значень становить

$$P_j(k, L_{bh}) = 2^{L_{bh}(1/k-1)}$$

Нехай $L_{bh} > L_{bm}$. Тоді ймовірність того, що Z вгадає $Hash(m)$ повідомлення m складе L_{bh}^{-1}. Ймовірність того, що Z обере для вгаданого h повідомлення, навіть за умов, що йому відомі усі пари

$$(m, Hash(m)) : P_v(L_{bh}, L_{bm}) = 2^{(L_{bm} - L_{bh})}$$

Дійсно, кількість геш значень для дійсних повідомлень складе $2^{L_{bm}}$, коли простір геш значень складе $2^{L_{bh}}$. Так як $L_{bh} > L_{bm}$, тоді

$$\forall h \in H : \exists m \in M : h = Hash(m).$$

Встановимо ймовірність того, що Z вгадає обране значення m, навіть так що $m \notin M$:

$$P_h(L_{bh}) = 2^{-L_{bh}}$$

Тепер можемо обрахувати складність екзистенційної та селективної підробки на схеми цифрового підпису. Для селективної підробки повідомлення m_1' з вірного повідомлення $\{m_1, m_2, \ldots m_k\}$ зловмиснику Z необхідно знайти таку пару (r', s'), що є вірним підписом для $\{m_1', m_2, \ldots m_k\}$.

Ймовірність селективної підробки є сумою ймовірностей вгадування значення у j повідомленні та ймовірності вгадування обраного значення. Ймовірність екзістенційної підробки є сумою ймовірностей вгадування значення у j повідомленні, ймовірності існування повідомлення для вгаданого геш значення та складності пошуку такого повідомлення. Складність пошуку повідомлення складає $2^{L_{bm}}$ обчислень/зберігань геш значень, у випадку використання ідеальної функції гешування. За парадоксом днів народження, кількість обчислень можна зменшити до $2^{L_{bm}/2}$.

Необхідно зауважити, що проведення атаки за парадоксом днів народження можливе тільки тоді, коли зловмисник має значення цифрового підпису для знайденого прообразу, та має сенс лише у разі високої активної власника ключа. Таким чином, за умови, що кількість повідомлень у множини, що підписуються дорівнює $k = const, L_{bm} = const$, безпечна кількість сформованих підписів дорівнюватиме $2^{L_{bm}/2}$. Надалі будемо розглядати екзистенційну підробку як атаку, коли Z може сформувати деякий вірний підпис (r', s) для випадкового повідомлення m з одного отриманого вірного підпису (r, s).

Тобто, враховуючи розрахунки та зауваження, ймовірність селективної підробки

$$P_s(k, L_{bh}) = 2^{-L_{bh}} P(k, L_{bh})$$

а ймовірність екзістенційної підробки в гіршому випадку

$$P_e(k, L_{bh}) = P(k, L_{bh}) \cdot 2^{L_{bm} - L_{bh}} \cdot 2^{-L_{bm}}$$

Легко побачити, що $P_e(k, L_{bh}, L_{bm}) = P_s(k, L_{bh})$. Надалі будемо використовувати $P_e(k, L_{bh})$.

Вирахуємо відповідну до атаки селективної підробки кількість бітів захисту:

$$\xi(k, L_{bh}) = \left\lceil log_2 \left(P_s(k, L_{bh}) \right)^{-1} \right\rceil. \quad (5)$$

Засновуясь на розрахунках, розглянемо можливість існування оптимальної схеми для скорочених повідомлень. Будемо вважати, що оптимальна схема існує, якщо для заданого рівня стійкості, що дорівнює Ξ бітам захисту, існує деяка функція $\mu(x)$, для якої справедливі наступні твердження:

$$\begin{cases} \psi(x) = \mu \circ Hash(x) \\ L_b(m_1 \| m_2 \| \ldots m_k) \geq \\ \quad \geq L_b\left(\psi(m_1) \| \psi(m_2) \ldots \| \psi(m_k)\right) + L_b(S). \end{cases} \quad (6)$$

Нескладно побачити, що якщо $\mu(x)$ існує та має наведені властивості, тоді Δ для схеми відповідає визначеному критерію (2).

Визначимо

$$\mu(x) = MSB(x, \varepsilon), \quad (7)$$

де $MSB(x, n)$ функція що вертає n старших бітів значення x та ε деяке таке значення, що

$$\forall m \in [m_1, m_2, \ldots m_k] : \varepsilon = \xi(k, L_b(\psi(m))) \geq \Xi. \quad (8)$$

Підставимо (6) до (8) та спростимо, враховуючи (7) та $\forall m \in \{m_1, m_2, \ldots m_k\} : L_b(m) = const$:

$$k(L_{bm} - \varepsilon) \geq L_b(S). \quad (9)$$

Знайдемо необхідну кількість повідомлень в множині, та ε, для створення ідеальної схеми (для випадку $L_{bm} = \Xi$):

$$\begin{cases} \xi(k, \varepsilon) \geq \Xi \\ k(\Xi - \varepsilon) \geq L_b(S) \end{cases} \quad (10)$$

Для підписів з доповненням, $L_b(S) = 4\Xi$.

Одним з рішень (10) є

$$\{k = 9, \varepsilon = \left\lceil ((\sqrt{65} - 7) * \Xi)^{-1} \right\rceil\}.$$

Наведемо приклад. Нехай $\Xi = 128$, тоді $L_b(S) = 2 * 256 = 512$, та $\varepsilon = 68$, $\lceil \xi(k, \varepsilon) \rceil = 128 = \Xi$. Безпечна кількість підписів становитиме 2^{64}.

Змінимо (10) для випадку $L_{bm} < \Xi$ (стисле повідомлення):

$$\begin{cases} \xi(k, \varepsilon) \geq \Xi \\ k(L_{bm} - \varepsilon) \geq L_b(S) \end{cases} \quad (11)$$

Наведемо приклад. Нехай $\Xi = 128$ та $L_{bm} = 80$. Обчислимо мінімальне k та ε такі, що кількість бітів захисту задовольняє Ξ.

$$\varepsilon = \left\lfloor L_{bm} - \frac{-\left(\sqrt{80\Xi^2 - 16 L_{bm}\Xi + {L_{bm}}^2} - 8\Xi - L_{bm}\right)}{2} \right\rfloor = 66$$

$$k = 4E(L_{bm} - \varepsilon)^{-1} = 8 \quad (12)$$

$$\xi(8, 80 - 14) = 130$$

Для створення ідеальної схеми з підпису з доданком з характеристиками $\Xi = 128$ та L_{bm} досить групи з $14 * 37 > 128 * 4 - 37$ повідомлень.

Зробимо висновок про існування схем, що задовольняють (2).

Необхідно зауважити, що у випадку, коли $L_{bm} < \Xi$, складність екзистенційної підробки становить не більше $2^{\varepsilon/2}$ та ймовірність проведення атаки для другої множини підписів $P_{e2}(\varepsilon, L_{bm}) = 2^{\varepsilon - L_{bm}}$.

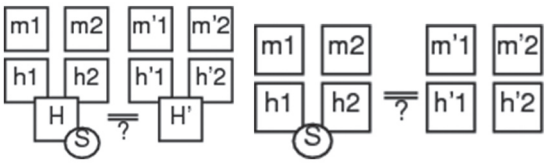
2. ЕЦП З ВІДНОВЛЕННЯМ ПОВІДОМЛЕННЯ

Загальновживаний метод вирішення поставленої задачі полягає у формуванні пакету

$\{M, H, S\}$, де $\forall m_i \in M, i \in [0,n): h_i = Hash(m_i)$, $S = Sign(H, K_a)$, та $Sign$ – алгоритм цифрового підпису з доданком. Схеми цифрового підпису, що базуються на перетвореннях у групі точок еліптичної кривої, мають бути побудовані над полем з бітовою довжиною у двічі більшою ніж заданий рівень стійкості, та зазвичай мають дві компоненти. Схеми гешування повинні мати вихід у двічі більший, ніж заданий рівень стійкості. Таким чином, в загальному випадку цифровий підпис має вдвічі більший обсяг ніж скорочене повідомлення.

Вирішити задачу більш ефективно можливо за рахунок використання підписів з відновленням повідомлення. Ці схеми більш відповідають встановленої семантиці.

Розглянемо схематичне зображення процесу підпису двох скорочених повідомлень. На рис. 1 зображено підпис з використанням підписів з доповненням. В процесі підпису формується нове геш значення H, що приймає участь в формуванні даних підпису. На відміну від схеми з доповненням, при використанні ЕЦП з відновленням (рис. 1 *а*) додаткове геш значення не формується: з підпису відновлюються безпосередні геш значення повідомлень, що потім мають бути перевірені окремо.



а *б*

Рис. 1. Підпис скорочених повідомлень з доповненням (*а*) та відновленням (*б*)

Схеми з відновленням повідомлення дозволяють скоротити $L_b(S)$. Це дозволяє зменшити кількість повідомлень в множині ідеальної схеми. Необхідно зауважити, що виграш можливо отримати тільки якщо множина геш-значень повідомлень буде цілком відновлена з підпису. В іншому випадку, схема зводиться до ЕЦП з доданком.

Повернемося до прикладу. Нехай $\Xi = 128$ та $L_{bm} = 80$. Обчислимо мінімальне k та ε такі, що кількість бітів захисту задовольняє Ξ для абстрактної схеми з відновленням повідомлення, що може відновлювати повідомлення з довільним L_{bh}.

$$\varepsilon = L_{bm} - \frac{-\left(\sqrt{24E^2 - 8L_{mb}E + L_{bm}{}^2} - 4E - L_{bm}\right)}{2} = 66$$
$$k = 2E(L_{bm} - \varepsilon)^{-1} = 18 \qquad (13)$$
$$\xi(18,66) = 128.$$

Для створення ідеальної схеми з підпису з доданком з характеристиками $\Xi = 128$ та L_{bm} досить групи з 19 повідомлень. Легко побачити, що використання підписів з відновленням повідомлення дозволяє створювати ідеальні схеми з кількістю повідомлень майже в двічі меншою.

Надалі будемо розглядати підписи з відновленням повідомлення.

3. КРИТЕРІЇ ВИБОРУ ОПТИМАЛЬНОЇ СХЕМИ ЕЦП ДЛЯ ПІДПИСУ МНОЖИН СКОРОЧЕНИХ ПОВІДОМЛЕНЬ

Задача підпису множин скорочених повідомлень є частковим випадком задачі підпису довільного повідомлення. Тому, довільні умовні та безумовні критерії оцінки перетворень ЕЦП з доданком можуть бути застосовані і для цього випадку.

Проаналізуємо п'ять алгоритмів, що засновано на перетвореннях у групі точок еліптичної кривої, з відновленням повідомлення стандарту ISO/IEC 9796-3 за наведеними критеріями. Надалі під підписами будемо розуміти ECNR, ECMR, ECAO, ECPV та ECKNR з стандарту ISO/IEC 9796-3.

Наведемо загальні безумовні та умовні критерії.

• *Експоненційна складність атаки повного, універсального розкриття.*

Усі підписи побудовані схемою подібною до класичної NR (наведено на рис. 1.б). Захист від атаки повного розкриття забезпечуються незворотнім перетворенням у групі точок еліптичної кривої та має експоненційну складність, для усіх підписів.

• *Практична захищеність схем ЕЦП від відомих атак.* Підписи з відновленням повідомлення базуються на схемі Ніберг-Рюпеля, та мають властивості подібні до інших підписів, що базуються на цій схемі. Таким чином, складність атаки повного розкриття та універсальної підробки еквівалентні ДСТУ 4145. Складність селективної та екзистенційної підробки підписів співпадає с задачею пошуку першого та другого прообразу геш значень повідомлень, що підписуються.

• *Відсутність слабких особистих ключів.* Для підписів з відновленням повідомлення дійсні дослідження стосовно слабких особистих ключів, що проведено до інших схем ЕЦП, заснованих за схемою Ніберг-Рюпеля. ДСТУ 4145 є однією з таких схем ЕЦП. Відомостей щодо слабких особистих ключів для ДСТУ 4145 немає.

• *Використання патентів на алгоритми в схемі ЕЦП, що передбачають ліцензування, та інші обмеження до застосування.* Згідно до звіту комітету JC-27:

– ECNR, ECKNR – не мають запатентованих алгоритмів

– ECPV – має запатентованих алгоритмів, але PVSSR, що є еквівалентом, запатентовано. Дозволяється вільне використання.

– ECAO – запатентовано (JP 3 434 251). Дозволяється вільне використання.

– ECMR – запатентовано (JP H09-160492). Дозволяється вільне використання.

• Відомий світовий досвід використання схеми ЕЦП. Стандартизація на державному та світовому рівнях. Відоме використання схем ЕЦП

з доданком, що засновуються на схемі Ніберг-Рюпеля – ДСТУ 4145. Використання схем з відновленням повідомлення не є розповсюдженою практикою.

– ECNR, ECKNR, ECMR, ECAO – прикладів використання не знайдено

– ECPV – знайдено приклади використання в виробах, що запатентовано в Сполучених Штатах Америки. Зазвичай – ігрові пристрої, засоби RFID. Приклади впровадження (за патентною базою США): US20100062844, US20080045342, US20110131401, US20110119474, US20080069347.

• *Часова складність обчислення підпису.* Доцільно порівнювати часову складність аналітично, використовуючи абстрактний результат для аналізу доцільності використання схеми ЕЦП на визначеному апаратному забезпеченні.

Для оцінки практичних показників схем ЕЦП з відновленням повідомлення було реалізовано макет. Експериментально встановлено, що часові характеристики схем ЕЦП не розрізняються. Найбільш витратні перетворення у групі точок еліптичної кривої, що займають 80% витраченого часу, та накладні операції – 10%. Якщо визначити показники як (E,S,O) як E – витрачено на перетворення у групі точок еліптичної кривої, S – витрачено на інші криптографічні перетворення, O – витрачено на взаємодію з ОС, та $E+S+O=1$, тоді

– $ECNR=(0.81,0.03,0.16)$

– $ECKNR=(0.82,0.06,0.13)$

– $ECAO=(0.81,0.033,0.153)$

– $ECMR=(0.81,0.029,0.16)$

– $ECPV=(0.81,0.02,0.17)$

• *Просторова складність обчислення підпису.* Просторова складність обчислення для всіх підписів знаходиться на одному рівні. Експериментально дослідити відмінність не вдалося.

Оберемо критерії, що є винятковими для схем ЕЦП підпису множин скорочених повідомлень:

• *Максимальний обсяг повідомлення, що відновлюється.* Покращення показника Δ можлива за рахунок скасування підсумкового гешування. Кількість гешів повідомлень, що можна відновити, зумовлює обмеження використання схеми на великих множинах скорочених повідомлень.

Для схем ЕЦП з відновленням з стандарту ISO/IEC 9796-3 дійсні наступні твердження:

– $ECNR, ECMR, ECKNR$ – $L_b(n-1)$, де n – бітова довжина поля ЕК

– $ECPV$ – довільна

• *Мінімальний обсяг повідомлення, що відновлюється.* Погіршує показник Δ за рахунок збільшення доданої частини повідомлення на малих множинах та повідомленнях.

Для схем ЕЦП з відновленням з стандарту ISO/IEC 9796-3 дійсні наступні твердження:

– $ECNR, ECMR, ECKNR$ – $L_b(n-1)$, де n – бітова довжина поля ЕК

– $ECPV$ – позначення обсягу – 2 байти

• *Можливість використовувати власний алгоритм формування відновлених даних.* Для підпису поодиноких повідомлень стандартами ЕЦП з відновленням повідомлень передбачається надання додаткової збитковості, для можливості перевірки підпису. В випадку вирішення задачі з підписом множини повідомлень та визначеними умовами, в визначенні додаткової збитковості немає необхідності -- повідомлення перевіряються окремо. Визначені алгоритми обчислення додаткової збитковості зменшують обсяг повідомлення, що можна відновити, та зменшують Δ.

– ECNR, ECKNR, ECMR – дозволяють використовувати довільні алгоритми формування частини що відновлюється.

– ECPV – додає значення довжини повідомлення до повідомлення що відновлюється

– ECAO – визначає алгоритм формування, але можливо використовувати схему з відсутньою додатковою збитковістю.

• Показник Δ

ECNR, ECMR, ECKNR, ECAO мають однакові обмеження щодо максимального/мінімального обсягу повідомлення. З (11) та (10) легко побачити, що не можливо створити ідеальну схему ЕЦП для підпису скорочених повідомлень за допомогою наведених алгоритмів -- загальний обсяг повідомлень, що необхідно утворювати для є значно більшою, ніж $n-1$ біт, де $n=2\Xi$, що можна відновлювати з цих підписів. Дійсно, якщо для створення ідеальної схеми необхідно знайти додаткові 2Ξ біт, та 2Ξ біт є максимальним обсягом, тоді кількість біт, що може бути використана для формування геш значеннь повідомлень становитиме $2\Xi-2\Xi=0$ бітів.

ECPV – не має обмежень на повідомлення що відновлюється. Згідно з (11) та (10) ідеальна схема для скорочених повідомлень може бути створена.

ВИСНОВКИ

В системах, де необхідно журналювати, або зберігати з інших причин отримані підписані повідомлення, постає питання ефективності використання простору.

В загальному випадку, ЕЦП електронного документу займає обсяг менший, ніж сам документ. У разі, коли документ є меншим за підпис не доцільно використовувати стандартні схеми ЕЦП з доданком. Використання ЕЦП з доданком збільшує загальний обсяг повідомлення з підписом більше ніж на 100%.

Запропоновано показник для схем ЕЦП – відсоток збільшення повідомлення. Запропоновано критерій ефективності та порівняння – збільшення повідомлення не має перевищувати 100%.

Запропоновано визначення скороченого повідомлення як такого, чий бітовий обсяг є

меншим, ніж бітовий обсяг геш значення необхідного для забезпечення визначеного рівня стійкості від селективної підробки.

Проаналізовано часний випадок - користувач ЕЦП підписує множину повідомлень скорочених повідомлень

Доведено можливість створення схем, збільшення повідомлень яких не перевищує 100%.

Визначено, що такі схеми можуть бути створені за допомогою ЕЦП з відновленням повідомлення стандарту ISO/IEC 9796-3 ECPV

Визначено, що серед інших підписів з відновленням повідомлення, що базуються на перетвореннях в групі точок ЕК, стандарту ISO/IEC 9796-3 немає таких, що мають переваги над ECPV

Пропонується використовувати ECPV для схем підписів скорочених повідомлень

Література

[1] ISO/IEC 9796-3: Discrete logarithm based mechanisms. – 2006

[2] Digital Signature Standart (DSS): FIPS 186-3

[3] Daniel R. L. Brown , Don B. Johnson, Formal Security Proofs for a Signature Scheme with Partial Message Recovery/ Lecture Notes in Computer Science.

Надійшла до редколегії 12.04.2012

Шевчук Олексій Анатолійович, аспірант кафедри БІТ, ХНУРЕ. Область наукових інтересів: захист інформації в ІТС.

УДК 681.3.07

Схемы ЭЦП для групп подписей маленьких сообщений / О.А. Шевчук // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 240–244.

Предлагается метод формирования ЭЦП для множеств маленьких сообщений с отсутствующей избыточностью. Предлагается критерий эффективности для подобных схем

Ключевые слова: ЭЦП, восстановление сообщения, оптимизация вычислений.

UDC 681.3.07

Digital signtature schemes for sets of small messages / O.A. Shevchuk // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 240–244.

The paper proposes the method of DS generation for sets of small messages with missing data redundancy and efficiency criteria for such schemes.

Keywords: DS, message recovery, optimization of calculations.

УДК 519.72:003.26

ОДНОНАПРАВЛЕННЫЕ ФУНКЦИИ С ИНФОРМАЦИОННО НЕВЫЧИСЛИМОЙ ЛАЗЕЙКОЙ

А.М. КУДИН

Рассматривается текущее состояние исследований в области построения однонаправленных функций и однонаправленных функций с лазейкой. Предлагается подход к построению однонаправленных функций с лазейкой, в котором лазейка неоднозначно связана с параметром функции и результатом вычисления функции.

Ключевые слова: теоретико-информационная стойкость криптосистем, стойкость асимметричных криптосистем, односторонние функции в криптографии, односторонние функции с потерей информации, общая теория оптимальных алгоритмов.

ВВЕДЕНИЕ

Одновременно с появлением идеи и первых реализаций принципа асимметричной криптографии начались исследования существования формального доказательства стойкости этих криптосистем к криптоанализу [1-15]. В отличие от симметричной криптографии, где существует прямая связь меры информативности шифртекста о ключе (открытом сообщении) [13] и криптографическими преобразованиями, в асимметричных криптосистемах эта статистическая мера информации с шифрующим преобразованием напрямую не связана. Достаточно быстро были установлены проблемы со стойкостью асимметричных криптосистем, следующих из этого факта: нестойкость систем при зашифровании источников открытого текста с малой энтропией, наличия информации об открытом тексте в шифртексте, вычислимой за полиномиальное время, алгоритмическая различимость за полиномиальное время шифртекстов, полученных от разных открытых текстов (для одного ключа зашифрования) [5]. Также было доказано [2] сводимость первых двух проблем к третьей и предложена новая концепция стойкости асимметричных криптосистем – концепция вероятностного шифрования, являющаяся вариантом применения метода рандомизации открытого текста перед зашифрованием. Стойкость при этом оценивалась через невозможность для вероятностного полиномиального алгоритма распознавания различных открытых текстов, соответствующих заданному шифртексту. Для введенного определения стойкости (в дальнейшем получившего абривиатуру PI (Polynomial Indistinguishability) стойкости) была получена нижняя граница увеличения длины открытого текста при рандомизации [4]. Недостатком данного подхода к оценке стойкости являлась сложность введения количественной меры, который удалось преодолеть, введя понятие семантической стойкости (абривеатура S (SemanticSecurity)) через модели теории игр. Количественной мерой стало так называемое «преимущество противника», определяемое через модуль разности вероятности случайного угадывания открытого текста и угадывания открытого текста по имеющимся шифртексту и открытому ключу. Дальнейшие исследования позволили установить взаимосвязь между обоими определениями и рассматривать стойкость относительно выбранного открытого текста (абривиатура CPA-стойкость (chosen plaintext security)). При этом оставалась проблема существования множества шифртекстов, для которых система не обеспечивала достаточную стойкость, т.е. определение стойкости было неравномерным относительно множества шифртекстов.

Более строгое и общее определение стойкости было введено относительно адаптивно выбранного шифртекста (абривиатура CCA-стойкость (chosen cipher text security)). Последнее определение позволило на основе вычислительной модели со случайным оракулом вплотную приблизиться к решению вопроса о существовании формального доказательства стойкости криптосистем «почти для всех» шифртекстов.

Все вышеперечисленные определения (PI, S, CPA, CCA-стойкости) оставались формальными моделями, не связанными напрямую с методами построения новых асимметричных криптосистем, удовлетворяющих этим определениям стойкости. Основной причиной этого явилось пересмотр не самого подхода к построению однонаправленных функций с лазейкой (ОДФЛ), а изменения структуры криптосистем. Заметим, что особенно характерно это проявилось при изучении стойкости криптосистем и протоколов типа «протоколов с неполным или нулевым разглашением секрета» [5]. Следствием этого явились появление новых моделей стойкости и требований к асимметричным криптографическим преобразованиям [3,4,6] (требования не сохранения гомоморфизма на множестве шифртекстов, требования анонимности ключа и т.п.) без пересмотра требований к ОДФЛ. Одной из первых идей, пересматривающими само построения ОДФЛ, явилась идея отказа от инъективности функции зашифрования, а также предложенные на этой основе ОДФЛ с потерей информации [8], вычислительная модель стойкости, основанная на использовании общей теории оптимальных алгоритмов [9-14].

В данной статье рассматриваются подходы к построению однонаправленных функций

с лазейкой для асимметричных криптосистем, стойких в теоретико-информационном смысле. Анализируются известные методы односторонних функций с лазейкой с потерей информации [8,16] и однонаправленные функции с информационно неопределенной лазейкой, предложенные автором [9-14].

1. КЛАССИЧЕСКИЕ ПОНЯТИЯ ОДНОСТОРОННИХ ФУНКЦИЙ И ИХ ПРИМЕНЕНИЕ В КРИПТОГРАФИИ

Введем некоторые определения, нужные для дальнейшего изложения.

Определение 1. Честная функция – функция $f:\{0,1\}^n \to \{0,1\}^{m(n)}$, если $n \le q(m(n))$, где $q(x)$ некоторый полином степени не выше k_0, для всех n. Степень полинома определяется вычислительными возможностями противника.

Определение 2.

Функция $\nu:\mathbb{N}\to\mathbb{R}$ называется «пренебрежимо малой функцией», если для $\forall c \ge 0$ существует k_c такое, что $\nu(k) < k^{-c}$ для всех $k \ge k_c$.

Определение 3.

Сильной односторонней функцией называется честная функция $f:\{0,1\}^n \to \{0,1\}^{m(n)}$, если:

1) Существует вероятностный алгоритм полиномиальной вычислительной сложности, вычисляющий $f(x)$ для $\forall x \in \{0,1\}^n$

2) для любого вероятностного алгоритма полиномиальной вычислительной сложности A существует пренебрежительно малая функция, что для всех $n \ge n_0$

$$P(f(z)=y; x \xleftarrow{R} \{0,1\}^n; y \leftarrow f(x); z \leftarrow A(1^n; y)) < \nu_A(n).$$

Определение 4.

Слабой односторонней функцией называется честная функция $f:\{0,1\}^n \to \{0,1\}^{m(n)}$, если:

1) Существует вероятностный алгоритм полиномиальной вычислительной сложности, вычисляющий $f(x)$ для $\forall x \in \{0,1\}^n$

2) для любого вероятностного алгоритма полиномиальной вычислительной сложности A существует полином p такой, что для всех $n \ge n_0$

$$P(f(z)\ne y; x \xleftarrow{R} \{0,1\}^n; y \leftarrow f(x); z \leftarrow A(1^n; y)) \ge 1/p(n).$$

Пример – умножение на множестве целых чисел (с обратной функцией факторизации). Грубая оценка доли чисел, являющихся произведением двух простых чисел приблизительно равного размера – $1/k^2$ (исходя из приблизительной вероятности того, что число длиной k бит будет простым $O(1/k)$).

Вообще говоря, имеет смысл говорить о семействе (множестве) односторонних функций с параметром длины входа, т.к. на практике конкретная длина входа зависит от возможностей вычисления обратной функции. Вместо определения множества функций можно определить адаптивный (в простейшем случае зависящий от размера входа) алгоритм вычисления обратной функции. Разница между двумя определениями в том, что для слабой односторонней функции мы требуем существования только некоторой не пренебрежительно малой части входов, на которых трудно вычислить обратную функцию. Для сильной требуем трудности вычисления обратной функции для всех, кроме незначительной части входов.

При этом существование слабых односторонних функций необходимо и достаточно для существования сильных. Кроме этого, известен простой метод [2], позволяющий строить сильные односторонние функции из слабых:

Пусть $f_1(x_1 \dots x_N) = f(x_1) \| \dots \| f(x_N)$ – сильная односторонняя функция, если $N = 2kp(k)$, $\forall x_i$ длиной k и $f(x_i)$ – слабые односторонние функции.

Заметим, что доказательство справедливо только для последовательной модели вычислений и только если информация, полученная при вычислении функции от одного аргумента не используется для вычислении функции от другого аргумента.

Определение 5.

Неравномерной (non-uniform) односторонней функцией называется функция

$f:\{0,1\}^n \to \{0,1\}^{m(n)}$, если:

1) Существует вероятностный алгоритм полиномиальной вычислительной сложности, вычисляющий $f(x)$ для $\forall x \in \{0,1\}^n$

2) для любого полиномиального по памяти алгоритма A и любого полинома q и всех $n > n_0$

$$P(f(z)\ne y; x \xleftarrow{R} \{0,1\}^n; y \leftarrow f(x); z \leftarrow A(y)) \ge 1/q(n).$$

Заметим, что поскольку для инвертирования функции используется алгоритм, полиномиальный по используемой памяти, то требования честности функции можно опустить.

Можно показать, что неравномерная односторонняя функция является сильной односторонней функцией.

Возможно, но не очень вероятно, что существует сильная односторонняя функция, которая не является неравномерной односторонней функцией.

2. ПОСТАНОВКА ЗАДАЧИ

Односторонние функции в криптографии рассматриваются под углом теории алгоритмов: не должно существовать эффективного в среднем (вероятностного) алгоритма криптоанализа по времени, но не по точности [14]. С другой стороны рассматривается недостаточность взаимной информации (даже уже в смысле Колмогорова) для восстановления одного слова по другому. Необходимо показать, что учет количества информации по Колмогорову об аргументе одностороннего криптографического преобразования с лазейкой позволяет увеличить эффективность криптосистемы, или с точки зрения «расширения» аргумента (увеличения его длины или с точки зрения длин ключей).

Такая постановка задачи связана с оценкой зависимости вида обратной функции от входных данных и оценки алгоритма вычисления обратной функции в случае **неравномерного** распределения входов. Вероятность успеха этого алгоритма зависит от распределения входов.

Фактически задача сводится к анализу распределения прообразов односторонней функции. При этом граничным случаем является равномерное распределение открытого текста (в случае его рандомизации или без избыточного кодирования). Но это является тривиальным случаем, поскольку при равномерном распределении открытого текста легко получить теоретико-информационную стойкость. С другой стороны, данный случай рассматривать не нужно, т.к. законный получатель тоже не получит никакой информации даже расшифровав сообщение. Естественно также не рассматривать случай, когда метод кодирования или параметры рандомизации являются долговременным ключом.

3. ОДНОСТОРОННИЕ ФУНКЦИИ С ПОТЕРЕЙ ИНФОРМАЦИИ

В модели lossy trapdoor function [8] (функции с потерей информации) неопределенность все равно вносится в открытый текст (функция становится не инъективной) путем предварительного шифрования (линейного, путем умножения на матрицу), а не в само шифрующее преобразование, как в вычислительной модели стойкости.

Основные отличия – 1) это конструкция «черного ящика» более эффективная, чем общая парадигма неинтерактивных нулевых знаний 2) эта конструкция позволяет строить стойкие в смысле атак по выбранному шифртексту (англ. CCA) криптосистемы основанные на **наихудшем случае** проблемы решеток.

Проблема в том, что достижения семантической стойкости в стандартной модели требуется внешний источник случайности (независимый от шифруемого сообщения). Главная проблема в том, что для инвертирования обратной функции требуется полный вход (т.е. случайность и шифртекст), а для расшифрования желательно, чтобы было достаточно только шифртекста. Интуитивно чувствуется, что нужна более сильная конструкция, чем семантическая стойкость. ОДФЛ с потерей информации (ОДФЛПИ) ведет себя одним из двух способов (неотличимых вероятностным полиномиальным алгоритмом друг от друга): первый – обычный, второй – размер образа существенно меньше, чем прообраза (например, размер входа n, размер образа $n/2$). Пусть

$$x \xleftarrow{random} \{0,1\}^n,\ c=(c_1,c_2)=(f(x),m\oplus h(x)).$$

Здесь $h(x)$ – ядро односторонней функции $f(x)$. Расшифрование: $m=c_2\oplus h\left(f^{-1}(c_1)\right)$.

Практически криптосистемы на основе ОДФЛПИ отличаются от предыдущих конструкций с рандомизацией открытого текста (например, вычислительной модели стойкости) фактически только тем, что открытый текст **не участвует в асимметричном шифровании**: практически вместо шифрования текста с помощью лазейки вырабатывается общая шифрующая гамма (или приводится к одному состоянию генератор псевдослучайной последовательности). При этом, поскольку размерность входных данных для генератора может быть меньшей, чем выхода, то при обратная функция является не инъективной, т.е. создается информационная неопределенность открытого текста и шифртекста.

Для реализациообщей схемы (framework) ОДФЛ с потерей информации можно использовать любую семантически стойкую криптосистему, обладающую несколькими специальными свойствами. Первое из таких свойств – система должна быть аддитивно гомоморфной, (*т.е. шифрование должно сохранятьоперацию сложения двух открытых текстов, т.е.* $(M_1\circ M_2\to C_1+C_2)$. Открытый текст рассматривается как n мерный вектор, перед шифрованием происходит предварительное кодирование ОТ («расширение ОТ») путем умножения его на матрицу ($M\cdot x$). Матрица должна быть обратимой, также выполняется ($I\cdot x=x$), для инъективной функции матрица ненулевая, для ОДФЛ с потерей информации – матрица нулевая. Дополнительно к этому криптосистема обладает двумя свойствами: остается стойкой при повторном использовании случайности для разных ключей, во-вторых гомоморфная операция изолирует случайность (т.е. случайность входного шифртекста зависит только от случайности выходного шифртекста). Концепция ОДФЛПИ может быть реализована на основании варианта системы Эль-Гамаля. Напомним, что зашифрование в криптосистеме Эль Гамаля при открытом ключе $h=\alpha^z \bmod p$ и секретном z осуществляется следующим образом: $r\in_R[1,p-1], c_1=\alpha^r \bmod p, c_2=x\cdot h^r \bmod p$. Обозначим как $E_h(x,r)$ зашифрование x при выбранном открытом ключе и случайном $r\in_R\{0,1\}$.

Расшифрование – $x=c_2\cdot(c_1^{\,z})^{-1} \bmod p$.

Вариант криптосистемы Эль-Гамаля, рассматриваемый в ОДФЛПИ отличается процессами зашифрования и расшифрования, а именно– зашифрование $c_1=\alpha^r \bmod p, c_2=\alpha^x\cdot h^r \bmod p$, расшифрование $x=\log_\alpha(c_2/c_1^{\,z})$. Здесь в качестве входных данных принимаются биты $x=\{0,1\}$ (или другой «небольшой» размер входных данных), поэтому вычисление дискретного логарифма производится путем простого перебора. Хорошо известно, что данный вариант криптосистемы обладает S стойкостью при условии сложности задачи распознавания Диффи-Хеллмана (DDH). Эта система является также аддитивно гомоморфной относительно операции:

$$E_h(x,r)\circ E_h(x',r')=E_h(x+x',r+r')$$

где операция $\circ$ означает покоординатное умножение шифртекстов и уязвимой относительно добавления величины $v\in Z_p$ даже без знания

открытого ключа. Рассмотренное определение односторонних функций с лазейкой и потерей информации не учитывает возможность введения неоднозначности (нарушение свойства инъективности отображения) в зависимость лазейки и открытых параметров для вычисления функции. Поэтому ниже рассматривается идея построения ОДФЛ, в которых алгоритм вычисления обратной функции без лазейки не просто **не реализуется вероятностным алгоритмом полиномиальной сложности**, **но не существует**.

4. ОДНОНАПРАВЛЕННЫЕ ФУНКЦИИ С ИНФОРМАЦИОННО НЕОПРЕДЕЛЕННОЙ ЛАЗЕЙКОЙ

Введем следующие обозначения. Пусть заданы множества X, Y. Пусть 2^Y – класс всех подмножеств множества Y. В работе [12] рассматривается оператор $S: X \times R_+ \to 2^Y$, где $R_+ = [0,\infty)$, называемый оператором решения и обладающий двумя свойствами:

$$S(x,0) \neq \varnothing, \forall x \in X,$$

$$\delta_1 \le \delta_2 \Rightarrow S(x,\delta_1) \subset S(x,\delta_2), \ \forall \delta_1, \delta_2 \in R_+, x \in X.$$

Для заданного $\varepsilon \ge 0$ элемент $y \in Y$, удовлетворяющий условию $y \in S(x,\varepsilon)$ называется ε – приближением. Задача поиска ε – приближения рассматривается при условии отсутствия полной (и, в общем случае точной) информации об элементе x, о котором известна некоторая информация $N(x)$, где: $N: X \to Y$ – информационный оператор в терминологии общей теории оптимальных алгоритмов, а Y – образ множества X. Зная $N(x)$ необходимо найти ε – приближение к x (рис. 1).

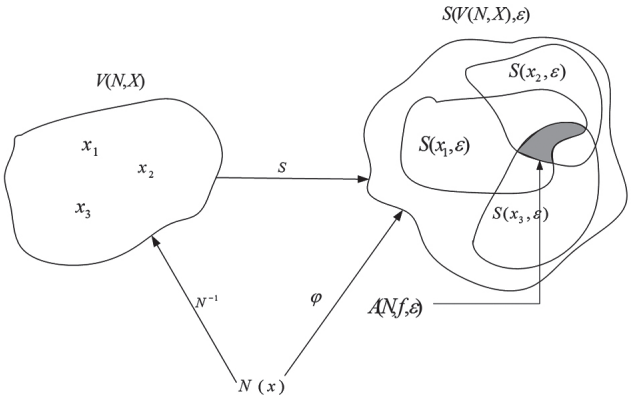


Рис. 1. Информационный оператор и оператор решения

Если множество

$$V(N,x) = \{\tilde{x} \in X : N(\tilde{x}) = N(x)\}$$

всех элементов $\tilde{x}$ неотличимых с помощью информационного оператора N от x состоит из одного элемента, то оператор N устанавливает взаимно-однозначное соответствие между множествами X и Y, и называется полным (и неполным в противном случае). Оператор решения, примененный к неполному информационному оператору, порождает множество

$$A(N,f,\varepsilon) = \bigcap_{\tilde{x} \in V(N,x)} S(\tilde{x},\varepsilon),$$

при этом для $\delta_1 \le \delta_2 \Rightarrow A(N,x,\delta_1) \subset A(N,x,\delta_2)$. Тогда величины

$$r(N,x) = \inf\{\delta : A(N,x,\delta) \neq \varnothing\}$$

$$\text{и } r(N) = \sup_{x \in X} r(N,x)$$

определяют нижние оценки точности решений, которые могут быть достигнуты при неполном информационном операторе.

В работе [15] доказано, что на классе идеальных алгоритмов

$$\Phi(N): N(x) \to G,$$

с введенными определениями локальной $e(\varphi,N,x) = \inf\{\delta : \varphi(N(x)) \in A(N,x,\delta)\}$ и глобальной $e(\varphi,N) = \sup_{x \in X} \varepsilon(\varphi,N,x)$ погрешностей информация $N(x)$ позволяет найти ε – приближение для произвольного $x \in X$ тогда и только тогда, когда выполняется одно из условий:

$$r(N) < \varepsilon,$$

$$r(N) = \varepsilon, \exists \varphi : \varphi(N(x)) \in S(x,\varepsilon(\varphi,N)), \forall x \in X.$$

В случае приближенной информации N_ρ (ρ – мера погрешности) результаты для нижних оценок определяются аналогично:

$$r(N_\rho) < \varepsilon,$$

$$r(N_\rho) = \varepsilon, \exists \varphi : \varphi(N_\rho(x)) \in S(x,\varepsilon(\varphi,N_\rho)),$$

$$\forall x \in X.$$

В отличии от точного информационного оператора, оператор N_ρ определяется через оператор информационной ошибки $E: H \times R_+ \to 2^H$, обладающий двумя свойствами:

$$E(h,0) = \{h\}, \forall h \in H,$$

$$\delta_1 \le \delta_2 \Rightarrow E(h,\delta_1) \subset E(h,\delta_2), \ \forall \delta_1, \delta_2 \in R_+, h \in H.$$

Приближенный оператор $N_\rho : X \to H$ удовлетворяет условию:

$$N_\rho(x) \in E(N(x),\rho), \forall x \in X.$$

Заметим, что если точный информационный оператор N неполон, то N_ρ тоже неполон, если же N полон, то N_ρ может оказаться как полным, так и неполным. Если оператор N_ρ полон, то $r(N_\rho) = 0$.

При построении ОДФЛ с использованием вышеописанного подхода отметим, что множество определения функции X может быть задано неполно и неточно. Тогда $N: X \to Y$ – информационный оператор, описывающий лазейку, без точности и полноты определения которого вычисление обратной функции, определенной оператором решения $S: X \times R_+ \to 2^G$ с необходимой точностью невозможно. Здесь G – множество оценок близости вычисленного значения обратной функции к «истинному», которое было задано. В зависимости от практической ситуации при построении асимметричной криптосистемы в качестве множества G могут

использоваться, например, апостериорные вероятности элементов множества X (как в теории информации Шеннона); множество предполагаемых открытых текстов или множество состояний конечного автомата, описывающего источник открытых сообщений X.

Выбором множества $\Phi(N(X))$ определяются вычислительные модели, которые могут быть использованы для вычисления обратной функции. При этом условие невозможности вычисления обратной функции без знания дополнительной информации о лазейке определяется как $r(N(X)) \geq \varepsilon > 0$, где $r(N(X))$ – радиус информации $N(X)$.

Особый интерес вызывает случай приближенной информации, т.е. сознательное внесение ошибок в процесс вычисления функции. При этом, как указывалось выше, при полном точном информационном операторе N оператор N_ρ может оказаться как полным, так и неполным.

В работе [14] приводится пример построения асимметричной криптосистемы, основанной на такой односторонней функции с информационно невычислимой лазейкой на базе модифицированной системы RSA.

Литература

[1] *G. Brassard* Relativized cryptography / IEEE Transactions on information theory. – V. IT-29. - Num.6. – 1983. – P. 877-890.

[2] *S. Goldwasser, S. Micali* Probabilistic Encryption / Journal of Computer and System Sciences. – №28, 1984. – P. 270-299.

[3] *D. Dolev, C. Dwork, M. Naor* Non-malleable cryptography / Proceedings of twenty-third annual ACM symposium on theory of computing. – New Orleans, Lousiana,Us,1991. – P. 542-552.

[4] *M. Bellare, P. Rogaway* Optimal asymmetric encryption / Advances in cryptology. – LCNCS V.950, 1994. – P. 92-111.

[5] *S. Goldwasser, M. Bellare* Lecture Notes on Cryptography. – Cambridge,Massachusetts, 2001. – 283 с.

[6] *M. Bellare, A. Boldyreva, A. Desai, D. Pointcheval* Key-privacy in public-key encryption. – ASIACRYPT 2001. – LCNS 2248. – pp. 566-582.

[7] *M. Abadi, P. Rogaway*. Reconciling two views of cryptography (The computational soundness of formal encryption) / Journal of Cryptology. – 2002. – Vol. 15. – № 2. – P. 103-127.

[8] *Chris Peikert Brent Waters* Lossy trapdoor functions and their applications Electronic Colloquium on Computational Complexity, Report No. 80 (2007)

[9] *Кудин А.М.* Оценка стойкости криптосистем с использованием Чебышевского радиуса информации / Искусственный интеллект. – № 4, 2002. – С. 568-573.

[10] *Кудин А.М.* Вычислительные модели стойкости криптосистем / Праці міжнародного симпозіуму «Питання оптимізації обчислень (ПОО-XXXIII)», присвяченого 50-річчю створення ІК ім. В.М. Глушкова НАН України. – К., 2007. – С. 150-152.

[11] *Кудин А.М.* Ограничения современных моделей описания криптосистем / Вісник Державного університету інформаційно-комунікаційних технологій. – Т. 6. – № 2. – 2008. – С. 144-146.

[12] *Кудін А.М.* Порівняльний аналіз математичних моделей стійкості криптосистем // Наукові вісті НТУУ «КПІ». – № 4 (72). – 2010. – С. 86-90.

[13] *Кудин А.М.* Криптографические преобразования нешенноновских источников информации // Кибернетика и системный анализ. – № 5. – 2010. – С.143-149.

[14] *Задирака В.К., Кудин А.М.* Анализ стойкости криптографических и стеганографических систем на основе общей теории оптимальных алгоритмов // JOURNAL OF QAFQAZ UNIVERSITY MathematisandComputerScience. – № 2. – 2010. – P.47-57.

[15] Д. *Трауб, Г. Васильковский, Х. Вожьняковский* Информация, неопределенность, сложность. – М.: Мир, 1988. – 184 с.

[16] *Л. Левин* Односторонние функции / Проблемы передачи информации. – 39(1), 2003.

Поступила в редколлегию 25.04.2012

Кудин Антон Михайлович, кандидат технических наук, старший научный сотрудник, доцент кафедры Физико-технического института Национального технического университета Украины «КПИ», докторант Института кибернетики им. В.М. Глушкова НАН Украины. Область научных интересов: теоретическая криптография, теория информации, основания асимметричной криптографии, методы реализации криптографических систем.

УДК 519.72:003.26

Односпрямовані функції з лазівкою, для обчислення якої не вистачає інформації / А.М. Кудін // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 245–249.

Розглядається сучасний стан досліджень в галузі побудови односпрямованих функцій та односпрямованих функцій із лазівкою. Пропонується підхід до побудови односпрямованих функцій із лазівкою, в якому лазівка неоднозначно пов'язана із параметром функції та результатом обчислення функції.

Ключові слова: теоретико-інформаційна стійкість криптосистем, стійкість асиметричних криптосистем, односпрямовані функції в криптографії, односпрямовані функції в криптографії з втратою інформації, загальна теорія оптимальних алгоритмів.

Іл. 01. Бібліогр.: 16 найм.

UDC 519.72:003.26

One-way functions with an informationally noncomputable trapdoor / A.M. Kudin // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 245–249.

The current state of researches of constructing one-way and trapdoor functions is considered. An approach is proposed to construct the trapdoor functions, in which the trapdoor is ambiguously related with a parameter of the function and function result.

Keywords: information theoretical security of cryptosystems, security of asymmetric cryptosystems, one-way functions in cryptography, lossy trapdoor functions, general theory of optimal algorithms.

Fig. 01. Ref.: 16 items.

БИОМЕТРИЧЕСКИЕ ИСТОЧНИКИ ИНФОРМАЦИИ, ИХ АНАЛИЗ И ПРИМЕНЕНИЕ

УДК 004.032.26

ПРОБЛЕМЫ КОМПРЕССИИ ДАННЫХ БОЛЬШОГО ОБЪЕМА В УСЛОВИЯХ НЕОПРЕДЕЛЕННОСТИ С ЦЕЛЬЮ ВЫЯВЛЕНИЯ ЛОКАЛЬНЫХ ОСОБЕННОСТЕЙ

Е.А. ВИНОКУРОВА

Предложено архитектуру и алгоритм обучения всех параметров вэйвлет-нейро-компрессора данных большого объема в условиях неопределенности с целью выявления локальных особенностей данных. Вэйвлет-нейро-компрессор позволяет сжимать не только данные, поданные в виде таблиц "объект-свойство", но и нестационарные нелинейные временные ряды в on-line режиме. Предложенный подход может быть использован для решения задач интеллектуальной обработки сигналов произвольной природы и в задачах аутентификации пользователей по их биометрическому образу.

Ключевые слова: сжатие данных большого объема, вэйвлет-нейро-компрессор, нестационарные временные ряды, аутентификация.

ВВЕДЕНИЕ

В настоящее время информация накапливается в огромные базы данных, объем которых измеряется в терабайтах. Фактически бесчисленное количество информации может получить кто угодно и где угодно через Интернет. Но для эффективного принятия решений полученную информацию необходимо обобщить и структурировать. Таким образом, когда количество данных, размерность и сложность скрытых зависимостей в них выше человеческих возможностей на первый план выходят методы интеллектуального анализа данных, которые позволяют извлечь локальные особенности и полезные знания.

Наряду с задачами, так или иначе связанными с проблемой аппроксимации, такими как прогнозирование, эмуляция, идентификация, распознавание образов, достаточно часто приходится решать задачу компрессии информации (сокращение размерности входного пространства признаков).

Задача компрессии данных широко применяется в различных приложениях биомедицины, техники, экономики, а также в задачах биометрической аутентификации личности, где необходимо выделить локальные особенности биометрических образов пользователя с целью занесения их в базу данных и для дальнейшего проведения аутентификации личности [1-5].

Выделим в качестве основных следующие типовые прикладные задачи снижения размерности анализируемого признакового пространства, которые рассматриваются в рамках многомерного анализа данных [6].

Первой задачей можно выделить отбор наиболее информативных признаков (включая выявление латентных факторов). Речь идет об отборе из исходного (априорного) множества признаков $X = (x_1, x_2, \ldots x_n)$, которые обладали бы свойством наибольшей информативности в смысле, определенном, как правило, некоторым специально подобранным для каждого конкретного типа задач критерием информативности. Также критерий информативности может быть нацелен на максимальную автоинформативность новой системы показателей, т.е. на максимально точное воспроизведение всех исходных признаков по сравнительно небольшому числу вспомогательных переменных.

Второй задачей является сжатие массивов обрабатываемой и хранимой информации. Такой вид задач связан с рассмотренной выше и, в частности, требует в качестве одного из основных приемов решения построения экономной системы вспомогательных признаков, обладающих наивысшей автоинформативностью. В действительности при решении достаточно серьезных задач сжатия больших массивов информации используется сочетание методов классификации и снижения размерности. Методы классификации позволяют перейти от массива, содержащего информацию по всем n статистически обследованным объектам, к соответствующей информации только по k эталонным образцам $k << n$, где в качестве эталонных образцов берутся специальным образом отобранные наиболее типичные представители классов, полученных в результате операции разбиения исходного множества объектов на однородные группы. Методы же снижения размерности позволяют заменить исходную систему показателей набором вспомогательных (наиболее автоинформативных) переменных.

Третьей задачей можно выделить визуализацию данных. Данная задача дает ответ на вопрос еще на предварительной стадии анализа данных – распадается ли анализируемая выборка на четко выраженные кластеры в заданном пространстве, каково примерное число их и т.д. Так как

максимальный размер фактически воспринимаемого пространства равен трем, поэтому естественно, возникает проблема проецирования анализируемых многомерных данных из исходного пространства на прямую, плоскость, поверхность, в крайнем случае – в трехмерное пространство так, чтобы интересующие специфические особенности исследуемой совокупности, если они присутствовали в исходном пространстве, сохранились бы и после проецирования. Следовательно, и здесь идет речь о снижении размерности анализируемого признакового пространства, но снижении, во-первых, подчиненном некоторым специальным критериям и, во-вторых, оговоренном условием, что размерность редуцированного пространства должна не превышать трех.

Еще одной задачей является сжатие временных рядов. Задача такого рода позволяет проводить компрессию многомерных временных рядов, во-первых, с целью анализа, прогноза, эмуляции уже объединенного процесса, который объединит все локальные особенности, а во-вторых, с целью упрощения хранения больших объемов многомерных временных рядов, это могут быть однородные процессы, которые снимаются с нескольких датчиков (такие как кардиограммы, энцефалограммы, голосовые наборы данных) или разнородные процессы, анализ которых может показать их зависимость между собой. Данная задача менее всех исследована, чем выше описанные, что подтверждает актуальность исследований в данном направлении.

Для решения рассмотренных задач компрессии существует ряд разработанных методов таких, как метод главных компонент [7], линейный дискриминантный анализ [8], вэйвлет-анализ [9-11], однако такие методы не могут быть применены для решения задач компрессии в реальном времени, с другой стороны, предложено ряд методов на основе нейронных сетей таких как нейронная сеть “Бутылочное горлышко” [12], нейронная сеть Хэбба-Сэнгера [13], нейронная сеть Оя-Карунена [14-17], нейронная сеть Рубнера-Шультена-Тэвена [18, 19], но все предложенные нейронные сети не могут применяться для компрессии временных рядов.

В работе [20-22] была предпринята попытка синтеза метода сжатия временных рядов с целью их дальнейшего сегментирования, но такой метод основан на методе главных компонент и не применим в on-line режиме.

1. АРХИТЕКТУРА ВЭЙВЛЕТ-НЕЙРО-КОМПРЕССОРА

В статье предлагается гибридная двуслойная вэйвлет-нейронная архитектура для решения задачи компрессии временных рядов и ее алгоритм обучения, сочетающие преимущества теории нейронных сетей и теории вэйвлетов, а именно способность обобщать и обучаться с возможностью выявления локальных особенностей.

Введем в рассмотрение двуслойную вэйвлет-нейронную архитектуру, представленную на рис. 1. Строительным элементом такой структуры является вэйвлет-нейрон [23] с нелинейными вэйвлет-синапсами.

При подаче на вход вэйвлет-нейрокомпрессора многомерного временного ряда $X=\{x^1(k),x^2(k),\ldots,x^n(k)\}$ на выходе сети получаем сигналы вида (так называемые главные компоненты)

$$y^m(k)=\sum_{i=1}^{n}\sum_{l=1}^{h_1}\varphi_{li}^{mi}(x^i(k))w_{li}^{mi}(k)\,, \qquad (1)$$

а соответственно сигналы, получаемые на выходе второго слоя, имеют вид

$$\begin{aligned}\hat{x}^i(k)&=\sum_{m=1}^{h}f_0^{im}\left(y^m(k)\right)=\\&=\sum_{m=1}^{h}\sum_{j=1}^{h_2}\varphi_{j0}^{im}\left(y^m(k)\right)w_{j0}^{im}(k)=\\&=\sum_{m=1}^{h}\sum_{j=1}^{h_2}\varphi_{j0}^{im}\left(\sum_{i=1}^{n}\sum_{l=1}^{h_1}\varphi_{li}^{mi}(x^i(k))w_{li}^{mi}(k)\right)w_{j0}^{im}(k),\end{aligned} \qquad (2)$$

где $\varphi_{li}^{mi}(\bullet)$, $\varphi_{j0}^{im}(\bullet)$ – вэйвлет-активационные функции первого и второго слоев соответственно, $w_{li}^{mi}(k)$, $w_{j0}^{im}(k)$ – синаптические веса первого и второго слоев соответственно, $y^m(k)$ – m-компонента сжатого многомерного временного ряда.

Двойной вэйвлет-нейрон состоит из двух слоев: скрытого слоя, в котором $2n$ вэйвлет-синапсов по h_1 вэйвлет-функций в каждом и выходного слоя, состоящего из $2n$ вэйвлет-синапса с h_2 вэйвлет-функциями.

В каждом вэйвлет-синапсе реализованы вэйвлеты, отличающиеся между собой параметрами центра и ширины, которые уточняются наряду с синаптическими весами с помощью тех или иных алгоритмов обучения.

С одной стороны, в качестве вэйвлет-активационных функций могут быть взяты различные семейства вэйвлетов, но с другой стороны, хорошо бы использовать адаптивную функцию принадлежности, параметры и форма, которой настраивалась бы в процессе обучения системы компрессии.

В данном случае предлагается использовать введенную нами [24, 25] настраиваемую активационную функцию, имеющую вид

$$\varphi_{li}^{mi}(x^i(k))=(1-\alpha_{li}^{mi}(\tau_{li}^{mi})^2)\exp\left(-\frac{(\tau_{li}^{mi})^2}{2}\right), \qquad (3)$$

$$\varphi_{j0}^{im}(y^m(k))=(1-\alpha_{j0}^{im}(\tau_{j0}^{im})^2)\exp\left(-\frac{(\tau_{j0}^{im})^2}{2}\right), \qquad (4)$$

где $\tau_{li}^{mi}(x^i(k))=(x^i(k)-c_{li}^{mi}(k))(\sigma_{li}^{mi}(k))^{-1}$,

$\tau_{j0}^{im}(y^m(k))=(y^m(k)-c_{j0}^{im}(k))(\sigma_{j0}^{im}(k))^{-1}$

α_{li}^{mi}, α_{j0}^{im} – настраиваемый параметр ($0\le\alpha\le1$).

Уточняемый параметр α позволяет настраивать форму активационной функции в процессе

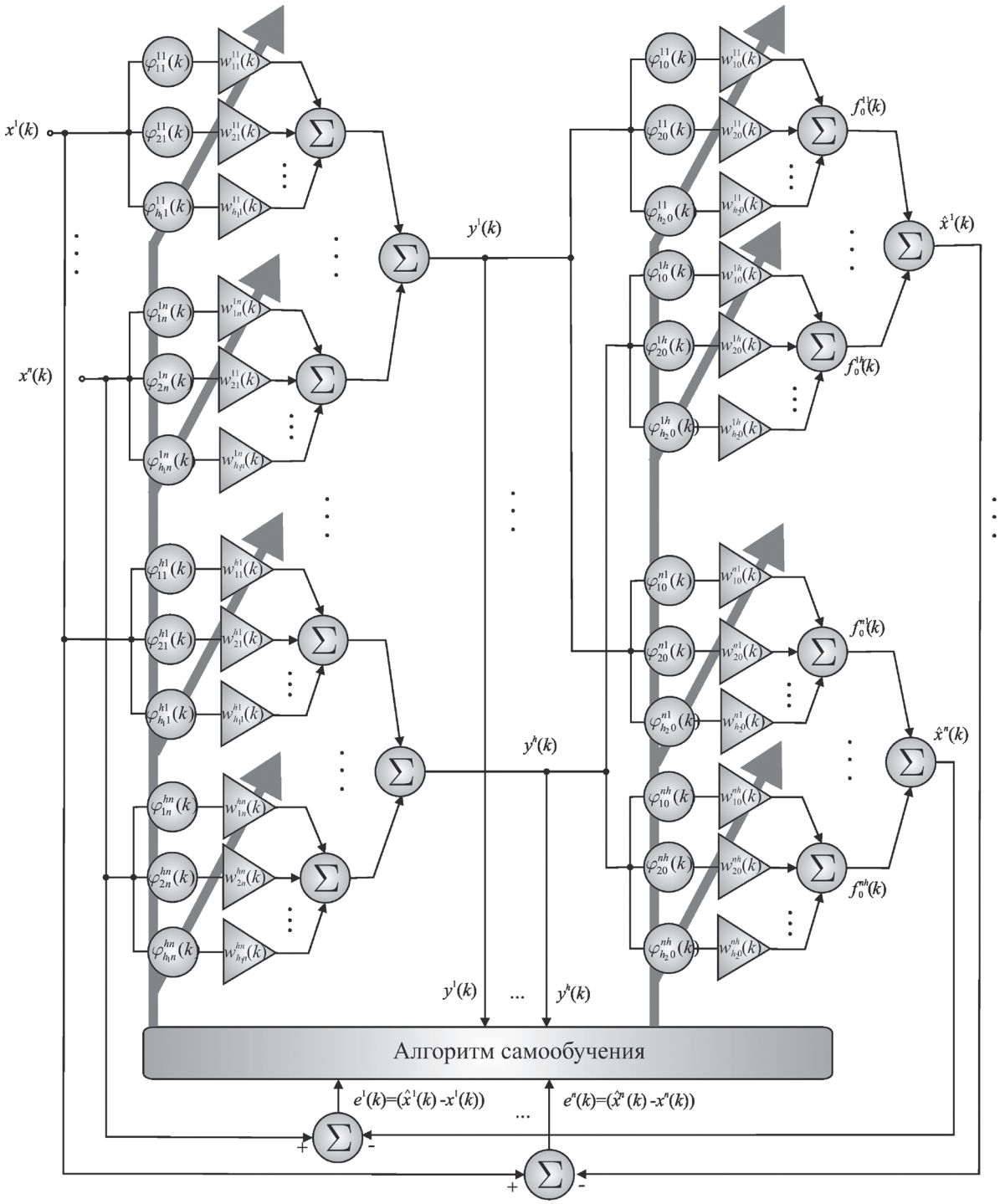


Рис. 1. Архитектура вэйвлет-нейро-компрессора

обучения составного адаптивного вэйвлона, при этом при $\alpha = 0$ получаем Гауссову функцию активации, при $\alpha = 1$ получаем вэйвлет-функцию «Mexican Hat», а при $0 < \alpha < 1$ – гибридную функцию активации. На рис. 2 приведены формы активационных функций в зависимости от параметра α.

3. АЛГОРИТМ ОБУЧЕНИЯ ВЭЙВЛЕТ-НЕЙРО-КОМПРЕССОРА

Базируясь на критерии обучения вида

$$E^i(k) = \frac{1}{2}\left(x^i(k) - \sum_{m=1}^{h} f_0^{im}(y^m(k)) \right)^2, \qquad (5)$$

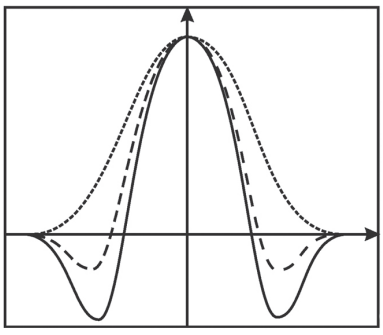


Рис. 2. Адаптивная функция активации с различными параметрами α (точечная линия $\alpha = 0$, пунктирная линия линия $\alpha = 0.5$, сплошная линия $\alpha = 1$)

можно записать алгоритм настройки синаптических весов и параметров вэйвлет-активационных функций первого слоя в виде

$$w_{li}^{mi}(k+1) = w_{li}^{mi}(k) + \\ +\eta^{w} e^{i}(k)\left[f_0^{im}(y^{m}(k))\right]' \varphi_{li}^{mi}(x^{i}(k)), \tag{6}$$

$$\begin{cases} c_{li}^{mi}(k+1) = c_{li}^{mi}(k) + \\ +\eta^{c} e^{i}(k)\left[f_0^{im}(y^{m}(k))\right]' w_{li}^{mi}(k)\dfrac{\partial\varphi_{li}^{mi}(x^{i}(k))}{\partial c_{li}^{mi}(k)}, \\ (\sigma_{li}^{mi})^{-1}(k+1) = (\sigma_{li}^{mi})^{-1}(k) + \\ +\eta^{\sigma} e^{i}(k)\left[f_0^{im}(y^{m}(k))\right]' w_{li}^{mi}(k)\dfrac{\partial\varphi_{li}^{mi}(x^{i}(k))}{\partial(\sigma_{li}^{mi})^{-1}(k)}, \\ \alpha_{li}^{mi}(k+1) = \alpha_{li}^{mi}(k) + \\ +\eta^{\alpha} e^{i}(k)\left[f_0^{im}(y^{m}(k))\right]' w_{li}^{mi}(k)\dfrac{\partial\varphi_{li}^{mi}(x^{i}(k))}{\partial\alpha_{li}^{mi}(k)}, \end{cases} \tag{7}$$

где $\eta^{w}, \eta^{c}, \eta^{\sigma}, \eta^{\alpha}$ – шаг алгоритма обучения,

$$\left[f_0^{im}(y^{m}(k))\right]' = \sum_{j=1}^{h_2} w_{j0}^{im}(k)\frac{\partial\varphi_{j0}^{im}(y^{m}(k))}{\partial y^{m}(k)}.$$

Алгоритм обучения второго слоя основывается на критерии, записанном в виде

$$E^{i}(k) = \frac{1}{2}(x^{i}(k) - \hat{x}^{i}(k))^2 = \frac{1}{2}(e^{i})^2(k), \tag{8}$$

где e^{i} – ошибка обучения.

Таким образом, алгоритм обучения синаптических весов и параметров активационных вэйвлет-функций второго слоя имеет вид

$$w_{j0}^{im}(k+1) = w_{j0}^{im}(k) + \eta_0 e^{i}(k)\varphi_{j0}^{im}(y^{m}(k)), \tag{9}$$

$$\begin{cases} c_{j0}^{im}(k+1) = c_{j0}^{im}(k) + \\ +\eta_0^{c} e^{i}(k) w_{j0}^{im}(k)\dfrac{\partial\varphi_{j0}^{im}(y^{m}(k))}{\partial c_{j0}^{im}(k)}, \\ (\sigma_{j0}^{im})^{-1}(k+1) = (\sigma_{j0}^{im})^{-1}(k) + \\ +\eta_0^{\sigma} e^{i}(k) w_{j0}^{im}(k)\dfrac{\partial\varphi_{j0}^{im}(y^{m}(k))}{\partial(\sigma_{j0}^{im})^{-1}(k)}, \\ \alpha_{j0}^{im}(k+1) = \alpha_{j0}^{im}(k) + \\ +\eta_0^{\alpha} e^{i}(k) w_{j0}^{im}(k)\dfrac{\partial\varphi_{j0}^{im}(y^{m}(k))}{\partial\alpha_{j0}^{im}(k)}, \end{cases} \tag{10}$$

где $\eta_0^{w}, \eta_0^{c}, \eta_0^{\sigma}, \eta_0^{\alpha}$ – шаг алгоритма обучения.

Таким образом, вэйвлет-нейро-компрессор позволяет реализовать сжатие и выявление локальных особенностей как данных представленных таблицей "объект-свойство", так и нестационарных нелинейных временных рядов в on-line режиме, что дает преимущество по сравнению с существующими методами.

Результаты экспериментов подтверждают эффективность предложенного вэйвлет-нейро-компрессора в решении задач биомедицинских приложений, в биометрических методах аутентификации пользователей методов, в анализе экономических показателей и других задачах.

ВЫВОДЫ

Предложена архитектура вэйвлет-нейро-компрессора и алгоритм его обучения всех параметров, обладающий следящими и фильтрующими свойствами. Предложенный подход позволяет решать задачу сжатия данных не только в виде таблице “объект-свойство”, но и многомерных нестационарных временных рядов произвольной природы с целью дальнейшей обработки.

Имитационные эксперименты подтверждают эффективность развиваемого подхода.

Литература

[1] *Fronthaler, H.* Local feature extraction in fingerprints by complex filtering / H. Fronthaler, K. Kollreider, J. Bigun // Proc. Intl. Workshop on Biometric Recognition Systems. – Dortmund: Gesundheit. – 2005. – P. 77–84.

[2] *Garcia-Salicetti, S.* BIOMET: A multimodal person authentication database including face, voice, fingerprint, hand and signature modalities / S. Garcia-Salicetti, C. Beumier, G. Chollet, B. Dorizzi, J. Lunter // Proc. International Conference on Audio- and Video-Based Biometric Person Authentication. – NY: Sears. – 2003. – P. 487-510.

[3] *Leung, M., Engeler, W., Frank, P.* Fingerprint image processing using neural network / M. Leung, W. Engeler, P. Frank // Proc. IEEE Region 10 Conf. on Computer and Comm. Systems. – NY: Sears. – 1999. – P. 657-714.

[4] *Kyong W.N.* A Feature Extraction Method for Binary Iris Code Construction / W.N. Kyong, L.Y. Kyong, S.B. Jun, S. Y. Woo // Proc. of the 2nd International Conference on Information Technology for Application (ICITA 2004). – 2005. – P. 210-220.

[5] *Connie T.* Palmprint Recognition with PCA and ICA, Image and Vision Computing / Connie T., Teoh, A., Goh, M. // New Zealand 2003, Palmerston North, New Zealand, (2003) 232–227.

[6] *Айвазян С.А.* Прикладная статистика. Классификация и снижение размерности / С.А. Айвазян. - Финансы и статистика. - 1989. - 608 с.

[7] *Лоули Д.* Факторный анализ как статистический метод / Лоули Д., Максвелл А. – М.: Мир, 1967. – 144 с.

[8] *Ким Дж.* Факторный, дискриминантный и кластерный анализ/ Дж. Ким, Ч.У. Мьюллер и др. — М.: Финансы и статистика, 1989. — 215с.

[9] *Chui C. K.* An Introduction to Wavelets / C. K. Chui. – New York: Academic, 1992. – 264 p.

[10] *Szu H.* Wavelet transforms and neural networks for compression and recognition / H. Szu, B. Telfer, J. Garcia // Neural Networks. – 1996. – 9. – P. 695-709.

[11] *Meyer Y.* Wavelets: Algorithms and Applications / Y. Meyer. – Philadelphia, PA: SIAM., 1993. – 133 p.

[12] *Cichocki A.* Neural Networks for Optimization and Signal Processing / Cichocki A., Unbehauen R. – Stuttgart: Teubner, 1993. – 526 p.

[13] *Sanger T.* Optimal unsupervised learning in a single-layer linear feedforward neural network / Sanger T. // Neural Networks. – 1989. – 2. – P. 459-473.

[14] *Oja E.* Neural networks, principal components, and subspaces / Oja E. // Int. J. of Neural Systems. – 1989. – 1. – P.61-68.

[15] *Oja E.* An analysis of convergence for a learning version of the subspace method / Oja E., Karhunen J. //J. Math. Anal. Appl. – 1983. – 91. – P.102-111.

[16] *Chen T.* Global convergence of Oja's subspace algorithm for principal component extraction / Chen T., Hua Y., Yan W.-Y. // IEEE Trans. on Neural Networks. – 1998. – 9. – P.58-67.

[17] *Бодянский Е.В.* Модифицированный нейрон Оя для анализа нестационарных данных / Бодянский Е.В., Плисс И.П., Тесленко Н.А. // Автоматизация: проблемы, идеи решения: Междунар. науч.-техн. конф.: тезисы докл. – Севастополь, 2006. – С.18-21.

[18] *Bishop C.M.* Neural Networks for Pattern Recognition / Bishop C. M. – Oxford: Clarendon Press, 1995. – 482 p.

[19] *Haykin S.* Neural Networks. A Comprehensive Foundation / Haykin S. – N.J.: Upper Saddle River, Prentice Hall, Inc., 1999. – 842 p.

[20] *Abonyi J, Feil B., Nemeth S.Z., Arva P.* Fuzzy Clustering Based Segmentation of Time-Series / Abonyi J, Feil B., Nemeth S.Z., Arva P. // Proc. 5th International Symposium on Intelligent Data Analysis. – Berlin, Germany. – 2003. – P. 275-285.

[21] *Abonyi J.* Introduction to Fuzzy Data Mining Methods. / Abonyi J, Feil B. // Handbook of Research on Fuzzy Information Processing in Databases / J. Galindo (Ed.) – 2008. – P. 55-95.

[22] *Abonyi J.* Cluster analysis for data mining and systems identification / Abonyi J, Feil B. – Birkhauser. Verlag AG. – Basel-Boston-Berlin. – 2007. – 303 p.

[23] *Bodyanskiy Ye.* An adaptive learning algorithm for a wavelet neural network / Bodyanskiy Ye., Lamonova N., Pliss I., Vynokurova O. // Blackwell Synergy: Expert Systems. – 22. – №5 – P. 235-240.

[24] *Бодянский Е.В.* Адаптивный вэйвлон и алгоритм его обучения / Бодянский Е.В., Винокурова Е.А.// Управляющие системы и машины. – 2009. – 1 (219). – С.47-53.

[25] *Bodyanskiy Ye.* Radial-basis-fuzzy-wavelet-neural network with adaptive activation-membership function / Bodyanskiy Ye., Vynokurova O., Yegorova E. // International Journal on Artificial Intelligence and Machine Learning. – 2008. – V.8. – II. – P. 9-15.

Поступила в редколлегию 15.03.2012

Винокурова Елена Анатольевна, кандидат технических наук, ведущий научный сотрудник Проблемной научно-исследовательской лаборатории АСУ, доцент кафедры Безопасности информационных технологий Харьковского национального университета радиоэлектроники. Область научных интересов: гибридные системы вычислительного интеллекта, вэйвлет-нейро-фаззи системы, прогнозирование, идентификация, компрессия, аутентификация на основе методов вычислительного интеллекта.

УДК 004.032.26

Проблеми стиснення даних великого обсягу за умов невизначеності з метою виявлення локальних особливостей / О. А. Винокурова // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 250–254.

Запропоновано архітектуру та алгоритм навчання усіх параметрів вейвлет-нейро-компресора даних великого обсягу за умов невизначеності з метою виявлення локальних особливостей даних. Вейвлет-нейро-компресор дозволяє стискати не тільки данні, що поданні у вигляді таблиці "об'єкт-властивість", але і нестаціонарні нелінійні часові ряди у on-line режимі. Запропонований підхід може бути використано для вирішення різних задач інтелектуальної обробки сигналів довільної природи та в задачах автентифікації користувачів за їх біометричним образом.

Ключові слова: стиснення даних великого обсягу, вейвлет-нейро-компресор, нестаціонарні часові ряди, автентифікація

Рис. 02. Библиогр.: 25назв.

UDC 004.032.26

Problems of mass data compression for the purpose of detecting local features under uncertainty conditions / O.A. Vinokurova // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 250–254.

An architecture and all parameters learning algorithm of a wavelet-neuro-compressor of mass data to detect local features under uncertainty conditions are proposed. The wavelet-neuro-compressor allows to compress not only data in an "object-property" table but non-stationary nonlinear time series in the on-line mode. The proposed approach can be used for solving of different problems of intelligent signal processing and in the tasks of authenticating users by their biometric image.

Keyword: mass data compression, wavelet-neuro-compressor, non-stationary time series, authentication.

Fig.: 02. Ref.: 25 items.

УДК 621. 391:519.2:519.7

МЕТОД ОЦЕНКИ ОТНОСИТЕЛЬНОЙ ЭНТРОПИИ И СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИСТОЧНИКОВ БИОМЕТРИЧЕСКОЙ ИНФОРМАЦИИ

И.Д. ГОРБЕНКО, И.В. ОЛЕШКО

В работе развивается подход к измерению информации, содержащейся в биометрической характеристике. Показывается, что особенности биометрической информации могут быть рассчитаны с помощью относительной энтропии. Производится сравнительный анализ источников биометрической информации.

Ключевые слова: биометрическая информация, относительная энтропия, биометрическая идентификация, матрица ковариации, метод главных компонент, Гауссово распределение.

ВВЕДЕНИЕ

Идентификация на основе биометрических данных – это средство автоматического опознавания личности на базе уникальных физических или поведенческих параметров. На сегодняшний день биометрические технологии идентификации личности получили широкое распространение в различных областях обеспечения безопасности: криминалистика; системы контроля доступа; системы идентификации личности; системы электронной коммерции; информационная безопасность (доступ в систему, авторизация на ПК); системы голосования, проведения электронных платежей; проекты государственной идентификации (пересечение границ, выдача виз) и т.д. Существует огромное количество методов биометрической идентификации. Актуальной остается проблема выбора того или иного метода. Сравнительный анализ методов биометрической аутентификации чаще всего производится на основании ошибок первого и второго рода. Использование относительной энтропии как критерия эффективности идентификации делает возможным не только сравнение биометрических признаков между собой, но и с персональным идентификационным номером (ПИН), паролем и другими методами аутентификации. Для сравнения систем биометрической аутентификации на основании критерия относительной энтропии необходимо определить количество информации каждой системы отдельно. Определим термин биометрическая информация следующим образом: уменьшение неопределенности идентичности человека за счет измерения набора биометрических характеристик. Основываясь на этом определении, в статье рассматривается алгоритм для измерения биометрической информации с помощью критерия относительной энтропии. Затем мы используем этот алгоритм для сравнительного анализа биометрической информации, содержащейся в различных алгоритмах распознавания по лицу и по радужной оболочке глаза.

1. АЛГОРИТМ ДЛЯ ВЫЧИСЛЕНИЯ БИОМЕТРИЧЕСКОЙ ИНФОРМАЦИИ

Алгоритм для вычисления биометрической информации с использованием относительной энтропии заключается в следующих шагах [1]: 1) выдвижение требований; 2) вычисление относительной энтропии биометрических характеристик; 3) Гауссова модель для вычисления биометрических характеристик и относительной энтропии; 4) методы регуляризации для вырожденных характеристик; 5) методы регуляризации для неполных данных.

Требования к особенностям биометрической информации:

1. Если распределение характеристик человека p равно распределению характеристик между людьми q, тогда нет информации, которая отличает человека и тогда биометрическая характеристика информации равна 0.
2. Если измерение особенности становится более точным, легче определить кого-то в популяции и биометрическая информация увеличивается.
3. Если у человека необычное значение характеристики (далекое от значения популяции), он становится более различимым, и информация о его биометрической характеристике увеличивается.
4. Биометрическая информация о некоррелированных характеристиках равна сумме биометрических информаций этих характеристик.
5. Характеристики, которые не связаны с определением идентичности, не повышают биометрическую информацию.
6. Биометрические характеристики, такие как вес и рост менее информативны.

Наиболее подходящая мера для характеристики биометрической информации – относительная энтропия $D(p\|q)$, где $p(x)$ и $q(x)$ распределения биометрических характеристик человека и населения соответственно. $D(p\|q)$ или расстояние Кульбака-Лейблера определяется как “дополнительные биты” информации необходимые для представления $p(x)$ относительно $q(x)$. $D(p\|q)$ определяется как:

$$D(p\|q)=\int_x p(x)\log_2\frac{p(x)}{q(x)}dx\,. \qquad (1)$$

Эту формулу можно объяснить следующим образом: относительная энтропия $D(p\|q)$

является дополнительной информацией, необходимой для описания распределения $p(x)$ на основании предполагаемого распределения $q(x)$. $D(p\|q)$ отличается от энтропии $H(p)$, которая является информацией, необходимой, в среднем, для описания особенности x с распределением $p(x)$. H является не подходящей величиной для измерения особенностей биометрической информации, т.к. она не учитывает, в какой степени каждый признак может идентифицировать человека. К примеру, характеристика, не относящаяся к идентичности – направление человеческого лица.

В общей биометрической системе, биометрические характеристики F измеряются для создания вектора биометрических характеристик $E(F\times 1)$ для каждого человека. Для человека p мы имеем N_p образцов характеристик, в то время как для населения – N_q образцов характеристик. Для удобства обозначения, мы сортируем измерения человека p так, чтобы они были первой группировкой населения. Определив x как значение случайной величины X, мы вычисляем среднее значение характеристик населения μ_q:

$$\mu_q = \underset{q}{E}[X] = \frac{1}{N_q}\sum_{i=1}^{N_q} x_i \,. \qquad (2)$$

Среднее значение характеристик человека p, μ_p, определяется аналогично, заменяя q на p.

Матрица ковариации характеристик населения Σ_q определяется следующим образом:

$$\Sigma_q = \frac{1}{N_q - 1}\sum_{i=1}^{N_q}(x_i - \mu_q)^t (x_i - \mu_q)\,. \qquad (3)$$

Матрица ковариации характеристик человека Σ_p определяется аналогично. Одной из важных общих сложностей с прямым измерением теоретической информацией является пригодность данных. Распределение трудно оценить точно, особенно на концах; для небольших $p(x)$ и $q(x)$, $\log_2(p(x)/q(x))$ будет иметь большое абсолютное значение. Обычно в таких случаях выполняется переход к модели с небольшим числом параметров. Гауссово распределение является наиболее общей моделью. На основании Гауссовой модели и соответствующих p и q, вычислим распределения биометрических характеристик человека и населения:

$$p(x) = \frac{1}{\sqrt{|2\pi\Sigma_p|}}\exp\left(-\frac{1}{2}(x-\mu_p)^t \Sigma_p^{-1}(x-\mu_p)\right), \qquad (4)$$

$$q(x) = \frac{1}{\sqrt{|2\pi\Sigma_q|}}\exp\left(-\frac{1}{2}(x-\mu_q)^t \Sigma_q^{-1}(x-\mu_q)\right). \qquad (5)$$

Из приведенных выше формул вычислим $D(p\|q)$:

$$D(p\|q) = \int p(x)(\log_2 p(x) - \log_2 q(x))dx =$$

$$= -k(\ln|2\pi\Sigma_p| - \ln|2\pi\Sigma_q| + 1 -$$

$$- \underset{p}{E}[(x-\mu_q)^t \Sigma_q^{-1}(x-\mu_q)]) =$$

$$= k\left(\ln\frac{|2\pi\Sigma_q|}{|2\pi\Sigma_p|} + trace((\Sigma_p + T)\Sigma_q^{-1} - I)\right) \qquad (6)$$

где $T = (\mu_p - \mu_q)^t(\mu_p - \mu_q)$ и $k = \log_2\sqrt{e}$. Это выражение вычисляет относительную энтропию для Гауссового распределения биометрических характеристик человека и населения. Оно соответствует большинству требований для измерения особенностей биометрической информации, представленных выше:

1. Если распределение характеристик человека соответствует распределению характеристик населения (p=q), то $D(p\|q)$=0, как и требовалось.

2. По мере улучшения измерения особенностей, значения матрицы ковариации Σ_p будут уменьшаться, приводя к уменьшению $|\Sigma_p|$ и увеличению $D(p\|q)$.

3. Если человек имеет значения характеристик далекие от средних по населению, T будет больше, и как результат $D(p\|q)$ будет больше.

4. Комбинация некоррелированных векторов характеристик дает сумму измерений человека $D(p\|q)$.

5. Добавление особенностей, не относящихся к идентичности, не будет изменять $D(p\|q)$.

6. Коррелированные характеристики менее информативны, чем некоррелированные. Они будут уменьшать Σ_p и Σ_q. Это приведет к понижению точности измерения $D(p\|q)$. В предельном случае для коррелированных характеристик, Σ_p станет единственной с нулевым детерминантом и $D(p\|q)$ будет неопределенно. Таким образом, наше измерение недостаточно в этом случае. Далее предлагается алгоритм для того, чтобы иметь дело с этим эффектом.

Рассмотрим методы регуляризации для вырожденных характеристик. Для защиты от численной нестабильности в наших измерениях мы хотим извлечь взаимно независимый набор G «важных» характеристик ($G \le F$). Для генерации отображения (U^t: $X \to Y$) из первоначальных биометрических характеристик $E(F\times 1)$ в новое пространство характеристик Y размером $G\times 1$ мы используем метод главных компонент (PCA) [2]. PCA может быть вычислен на основании декомпозиции единственного значения (SVD) [3] матрицы ковариационных характеристик следующим образом:

$$US_qU^t = svd(\mathrm{cov}(X)) = svd(\Sigma_q)\,, \qquad (7)$$

где Σ_q – положительная матрица, U – ортонормированная матрица, S_q – диагональная матрица. Мы выбираем для выполнения PCA распределение характеристик населения, а не человека, так как оно основано на гораздо большем количестве данных, и поэтому, вероятно, будет более надежной оценкой. Значения S_q определяют значимость

каждой характеристики в пространстве PCA. Характеристика j с небольшим $[S_q]_{j,j}$ будет оказывать незначительный эффект на всю характеристику биометрической информации. Мы используем этот анализ для упорядочивания Σ_q и для отказа от вырожденных характеристик при помощи исключения SVD. Выбирается округленный порог j, где $[S_q]_{j,j} < 10^{-10}[S_q]_{1,1}$. Основываясь на этом пороге, S_q усекается до $G \times G$, а U усекается до $F \times G$. Используя за основу U, мы раскладываем на составные части личностную ковариантность в пространстве характеристик Y:

$$S_p = U^t \Sigma_p U , \quad (8)$$

где S_p – необязательная диагональная матрица. Основываясь на этих регуляризационных схемах, перепишем $D(p \| q)$ в PCA пространстве:

$$D(p \| q) = k(\beta + trace U((S_p + S_t)S_q^{-1} - I)U^t) , \quad (9)$$

где $\beta = \ln \frac{|S_q|}{|S_p|}$ и $S_t = U^t T U$.

Рассмотрим регуляризационный метод для неполных данных. Выражение, разработанное выше, решает проблему некорректности Σ_q. Тем не менее, Σ_q может оставаться единственным в общем случае, когда только небольшое число образцов каждой личности доступно. Пусть даны N_p изображений личности, из которых вычислены G характеристик, Σ_q будет единственной, если $G \geq N_p$. На практике этот случай является общим, поскольку большинство биометрических систем вычисляют много сотен характеристик и редко бывает более 10-ти образцов каждого человека. Чтобы побороть эту проблему, мы разработали оценку, которая может служить нижней границей. Чтобы сделать это, мы делаем следующие предположения:

1. Оценки дисперсии характеристик матрицы $[S_p]_{i,j}$ верны для всех i.
2. Оценки ковариации характеристик матрицы $[S_p]_{i,j}$ для ij верны только для наиболее важных L характеристик, где $L < N_p$.

Характеристики, которые не считаются правильными, основываясь на этих предположениях, устанавливаются в 0 умножением S_q на маску M, где

$$M = \begin{cases} 1, \text{если } i = j \text{ или} (i < L \text{ и } j < L) \\ 0, \text{в остальных случаях} \end{cases} \quad (10)$$

Это выражение регуляризирует матрицу ковариации характеристик человека Σ_p и гарантирует, что $D(p \| q)$ не отклоняется. Чтобы выяснить влияние регуляризации на $D(p \| q)$, мы отмечаем, что ковариация характеристик человека $|\Sigma_p|$ будет уменьшаться до 0, что приведет к тому, что оценка дифференциальной энтропии уйдет в ∞. Мы рассматриваем эту регуляризационную стратегию для создания нижней границы биометрической характеристики информации. Выбор L представляет собой компромисс между использованием всех имеющихся измерений (с использованием большого L) и избеганием численной нестабильности, когда S_p близко к сингулярному (с использованием небольшого L).

3. ВЫЧИСЛЕНИЕ БИОМЕТРИЧЕСКОЙ ИНФОРМАЦИИ ЛИЦА

Вычисление биометрической информации лица осуществляется с использованием описанного метода. Чтобы протестировать алгоритм необходимо иметь несколько изображений одной личности. Используя базу данных по лицу Aberdeen [4] мы выбираем 18 фронтальных изображений каждого из 16 человек, для которых мы вычисляем PCA компоненты лица, используя алгоритм [5] и характеристики линейной дискриминанты Фишера (FLD), используя алгоритм, описанный в [6]. Первоначально все изображения лиц были зарегистрированы во вращении и определен их масштаб для того, чтобы знать позицию глаз в точках (50,90) и (100,90). Далее изображения были обрезаны до размера 150x200 пикселей, выровнены их гистограммы для покрытия диапазон интенсивности 0-255. Биометрические характеристики были вычислены из набора N_q изображений, используя различные методы компонентного анализа такие как PCA и FLD. μ_p и μ_q – $F \times 1$ вектора среднего распределения характеристик населения и человека, Σ_p и Σ_q – матрицы ковариации характеристик человека и населения соответственно.

Процесс декомпозиции характеристик был проведен на 18 изображениях каждого из 16 человек (всего 288 изображений). Для декомпозиции характеристик PCA и FLD было вычислено 288 отдельных векторов и 100 наиболее значимых компонент использовались для дальнейших анализов. На рис. 1 проиллюстрированы главные компоненты лица PCA. Слева направо показаны компоненты под номерами 3,15,35,55.

Рис. 1. Главные компоненты лица

На рис. 2 проиллюстрированы базисные дискриминанты лица FLD. Слева направо показаны дискриминанты под номерами 7, 10, 30, 50.

Рис. 2. Базисные дискриминанты лица

Используя FLD и PCA компоненты, для каждого из 16 человек рассчитано $D(p\|q)$, используя выражение (9). Принимается, что p и q имеют Гауссово распределение. Чтобы проверить достоверность Гауссовой модели для наших данных мы используем следующие тесты нормальности:

• тест Колмогорова-Смирнова: сравнение распределений значений в 2-х векторах данных X_1 и X_2, где X_1 представляет собой случайную выборку из исходного распределения, а X_2 – следующее идеальное гауссово с нулевым средним и дисперсией. Нулевая гипотеза состоит в том, что X_1 и X_2 взяты из того же непрерывного нормального распределения. Мы отвергаем нулевую гипотезу при $p<0.01$.

• тест Лилифора [7]: оценивает гипотезу, что x имеет нормальное распределение с неопределенным средним и дисперсией, против альтернативы, что X не имеет нормального распределения. Этот тест сравнивает эмпирическое распределение X с нормальным распределением, имеющим такое же среднее и дисперсию как X. Мы отвергаем нулевую гипотезу при $p<0{,}01$.

Используя эти тесты, получаем в среднем 88% и 89% для FLD и PCA компонент предельного распределения распределены нормально.

После подгонки распределений $p(x)$ и $q(x)$ к гауссовой модели, мы анализируем биометрическую информацию для FLD и PCA методов. PCA компоненты показаны на рис. 3. Из него видно постепенное снижение биометрической информации после 2-ой главной компоненты.

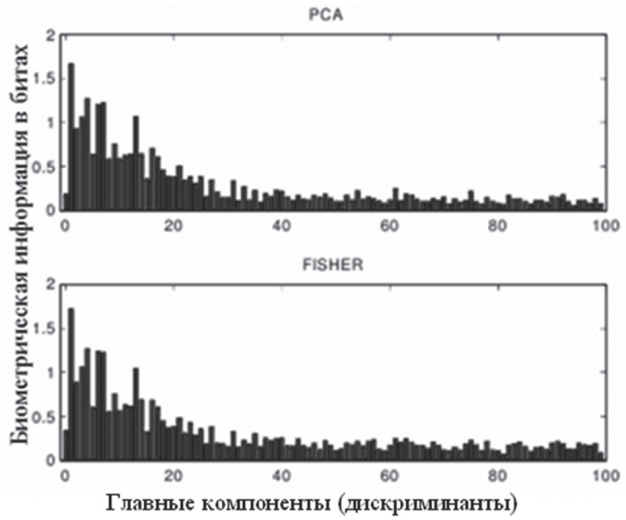
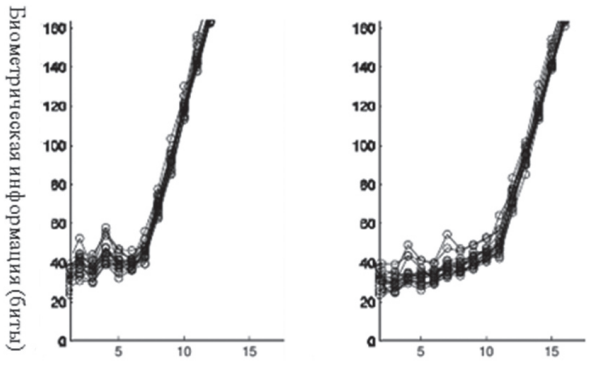


Рис. 3. Зависимость биометрической информации от номера компоненты для PCA и FLD

Такую форму графика можно объяснить природой PCA декомпозиции: чем выше номер главной компоненты, тем более высокие частоты деталей. Так как шум увеличивается с частотой, биометрическая информация при более высоких номерах PCA компонент будет меньше. Сумма биометрической информации за первые 100 PCA главных компонент для одного человека равна 40,5 бит. Биометрическая информация, вычисленная с использованием FLD дискриминант, кажется похожей на PCA. Для FLD дискриминант наибольшая биометрическая информация характерна для доминирующих ‘лиц Фишера’. Поскольку 18 изображений одного человека используются для вычисления матрицы ковариации, попытка вычислить $D(p\|q)$ для более чем 17 характеристик потерпит неудачу, т.к. Σ_p единственная. Относительно небольшое значение S_q показывает, что нет вырожденных характеристик для PCA и FLD алгоритмов. Как бы то ни было, S_p плохо обусловлено. Для преодоления этого мы предлагаем регуляризационную схему (формула 10), основанную на маске с точкой отсечения L. Эта схема основана на диагональной структуре S_p, как показано на рис. 4.

Рис. 4. Регуляризированная ковариационная матрица S_p с доминирующими компонентами вдоль диагонали

Для гарантирования сходимости размер маски L устанавливается значением меньшим, чем N_p. Мы решаем эту сингулярность уравнения (9), используя маску для S_p, основанную на параметре L. Для дальнейшего исследования воздействия параметров L и N_p мы искусственно уменьшаем N_p с помощью случайного удаления некоторых изображений личности. Результат для PCA компонент каждого человека как функция от L показан на рис.5.



а *б*

Рис. 5. Биометрическая информация (в битах) относительно размера маски L

Кривая напоминает форму клюшки. На рис. *а* и *б* представлены значения для различных N_p (изображений одного человека): на рис. *а* – N_p=8, на рис. *б* – N_p=12. Кривая показывает, что

$D(p\|q)$ необоснованно увеличивается, когда Σ_p становится сингулярным ($L \geq N_p$). Очевидно, что значения $D(p\|q)$, находящиеся выше "колена клюшки" не являются достоверными. Относительная энтропия увеличивается с размером маски. При приближении значений L к N_p происходит переоценивание $D(p\|q)$. С другой стороны малые значения L будут приводить к недооцениванию $D(p\|q)$. Чтобы получить точную оценку $D(p\|q)$ необходимо идти на компромисс между этими результатами. Мы выбрали L=3/4 N_p. Используя алгоритм, изложенный выше, и значение L мы вычисляем общую биометрическую информацию для различных алгоритмов распознавания по лицу. Для PCA компонент средняя биометрическая информация $D(p\|q)$ = 45 бит, а для FLD дискриминант $D(p\|q)$ = 37 бит. Если PCA и FLD компоненты объединить (сделать 200 характеристик) средняя биометрическая информация $D(p\|q)$ =55,6 бит. Биометрическая информация для FLD дискриминант меньше, чем для PCA компонент. Это можно объяснить тем, что PCA главные компоненты содержат информацию о выражении лица и освещенности. Большее значение биометрической информации указывает на то, что набор характеристик, использующийся в биометрической системе, содержит больше различающей информации, что должно привести в итоге к снижению ошибок первого и второго рода.

4. ВЫЧИСЛЕНИЕ БИОМЕТРИЧЕСКОЙ ИНФОРМАЦИИ РАДУЖНОЙ ОБОЛОЧКИ ГЛАЗА

Методы идентификации личности по радужной оболочке построены по одному и тому же принципу – выделение частотной или какой-либо другой информации о текстуре радужки из изображения и сохранение этой информации в виде специального кода (для системы Daugman этот код получил специальное название – IrisCode (радужковый код)). Построение кода производится в три этапа:

1. Выделение <баранки> радужки из общего изображения

2. Предобработка полученного изображения – например убирание шума (denoising), улучшение изображения (enhansing), в том числе выравнивание гистограммы, убирание блика. Некоторые методы "разворачивают" круглый зрачок в прямоугольное изображение – происходит переход из полярных координат в декартовы. Иногда после такой "развертки" часть изображения отрезается, чтобы накопленная на данном этапе ошибка не повлияла на качество распознавания.

3. Составление кода. Предварительно обработанное изображение фильтруется способом, зависящим от конкретного метода. По результатам фильтрации составляется представление в виде кода.

Для кодов необходимо выработать критерий сравнения. Часто код записывается в виде последовательности битов и критерием сравнения служит код Хэмминга. В частности, код Хэмминга используется в системах Daugman, Tisse [8]. Наглядно система аутентификации по радужной оболочке приведена на рис. 6.

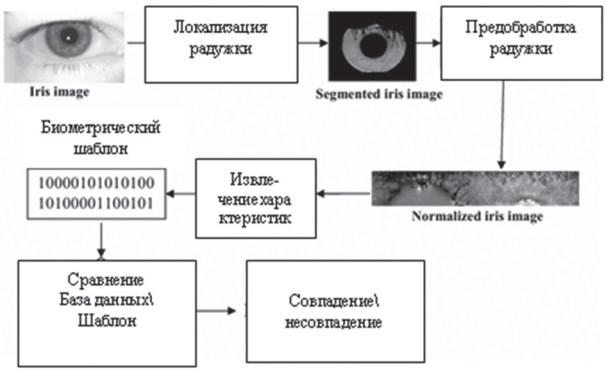


Рис. 6. Различные этапы в системе распознавания по радужке

Для вычисления биометрической информации радужной оболочки глаза использовалась база данных CASIA, которая содержит 689 изображений радужных оболочек глаз, взятых у 108 людей (6-7 изображений каждого человека). Изображение радужной оболочки было предобработано. Далее были вычислены PCA главные компоненты, используя алгоритм, описаный в [5] и ICA (Independent Component Analysis) компоненты, используя алгоритм, описанный в [6]. Для PCA метода и метода независимых компонент (ICA) было вычислено 327 векторов характеристик, которые были использованы для дальнейшего анализа. На рисунках 7 и 8 проиллюстрирована биометрическая информация, вычисленная для PCA и ICA компонент радужки. На рисунке 7 показана биометрическая информация, вычисленная для 327 PCA компонент. Верхний график вычислен для алгоритма Masek, а нижний – с использованием расширенной техники. Стандартное отклонение показано внизу каждого графика.

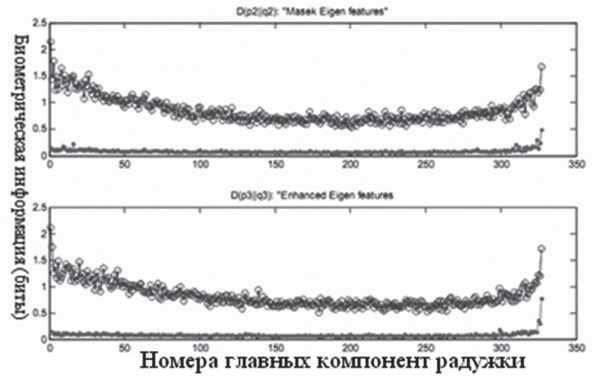


Рис. 7. Зависимость биометрической информации радужки от номера PCA компоненты

На рис. 8 показана биометрическая информация, вычисленная для 327 ICA компонент.

Верхний график вычислен для алгоритма Masek, а нижний – с использованием расширенной техники. Стандартное отклонение изображено внизу каждого графика.

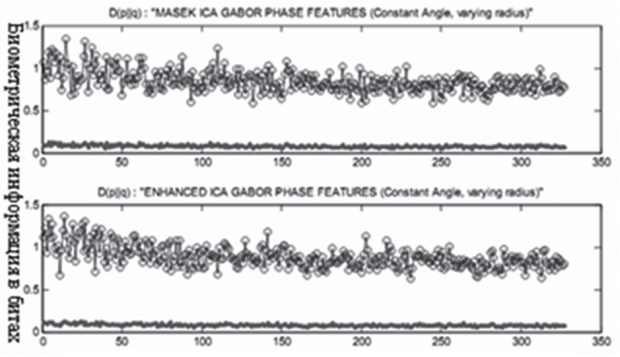


Рис. 8. Зависимость биометрической информации радужки от номера ICA компоненты

Используя алгоритм для вычисления биометрической информации, описанный выше, и базу данных CASIA, получаем 278 бит биометрической информации для PCA главных компонент радужки вычисленных по алгоритму Masek и 267 бит биометрической информации при использовании расширенной техники. Разницу в биометрической информации можно объяснить тем фактом, что алгоритм Masek не полностью устраняет ресницы из области радужки и эти пиксели ложно используются как информация о радужке. Это повышает биометрическую информацию, т.к. алгоритм принимает ресницы за текстуру радужки, делая ее тем самым более отличимой в наборе. Для ICA компонент средняя биометрическая информация составила 288 бит, используя сегментационный алгоритм Masek и 277 бит, используя расширенную сегментационную технику. Как видим, количество биометрической информации для ICA и PCA компонент, очень близко. ICA компоненты содержат больше информации, т.к. они соответствуют модели характеристических данных радужки лучше.

5. СРАВНИТЕЛЬНЫЙ АНАЛИЗ ИСТОЧНИКОВ БИОМЕТРИЧЕСКОЙ ИНФОРМАЦИИ

Сравнительный анализ источников биометрической информации производился на основании критерия относительной энтропии. Сравнение производилось по следующим биометрическим источникам: изображение лица, радужная оболочка глаза. Выделение компонент из изображения лица происходило с использованием алгоритмов PCA, FLD и ICA, а из радужной оболочки глаза с помощью - ICA и PCA. Результаты сравнений приведены в табл. 1.

Исходя из таблицы, можно сделать вывод о том, радужная оболочка глаза содержит больше биометрической (различающей) информации, чем изображение лица и, соответственно, аутентификация на ее основе будет более надежной.

Таблица 1

Сравнительный анализ источников биометрической информации

Алгоритм	Биометрическая информация (биты)	
	Лицо	Радужка
PCA	45	278
ICA	39	288
FLD	37	

ВЫВОДЫ

В данное время биометрические технологии идентификации личности получили широкое распространение. Актуальной является задача выбора того или иного метода. Чаще всего производится сравнение биометрических методов аутентификации на основании ошибок первого и второго рода. В работе предлагается производить сравнительный анализ источников биометрической информации на основании критерия относительной энтропии. Сравнительный анализ производился по 2-м источникам биометрической информации: радужная оболочка глаза и изображение лица. Выделение компонент лица осуществлялось с использованием методов компонентного анализа PCA, FLD и ICA. Выделение компонент радужки производилось с помощью метода PCA и метода независимых компонент.

Проблемным вопросом в нашей работе было: как определить распределение биометрических характеристик для населения. Мы использовали типичный подход: приняли нашу базу данных за адекватное представление населения.

С использованием описанного нами метода, была вычислена относительная энтропия для лица: для метода PCA она составила 45 бит, для FLD – 37 бит, а для ICA – 39 бит. Количество биометрической информации радужной оболочки глаза составило: для PCA метода 278 бит, а для ICA – 288 бит. Такие результаты совместимы с предыдущими исследованиями радужной оболочки глаза. Так, Daugman заявлял [9], что комбинаторная сложность фазовой информации радужной оболочки глаза различных людей составляет около 249 степеней свободы. Cover и Thomas [10], используя радужку диаметром в 11мм, просчитали, что ее биометрическая информация составляет 241 бит. Различие наших данных со значениями Daugman и Thomas можно объяснить тем, что диаметр радужной оболочки в нашей системе принимал значения от 11 до 11,5мм. А разница в 0,5мм дает увеличение биометрической информации на 28,52 бита.

На основании проведенного нами анализа, можно сделать вывод о том, что радужная оболочка глаза содержит больше биометрической информации, чем изображение лица. А это означает, что набор характеристик, использующийся в системе распознавания по радужке, содержит больше различающей информации, что должно

привести в итоге к снижению ошибок первого и второго рода.

Использование относительной энтропии как критерия сравнительного анализа позволяет сравнивать не только методы биометрической аутентификации между собой, но и с ПИНом, паролем и с другими методами.

Литература

[1] *Adler A.* Towards a measure of biometric feature information / A. Adler, R. Youmaran, S. Loyka // Pattern Anal. Appl. – 2009. – №12(3). – P. 261-270.

[2] *Draper, B.A., Baek, K., Bartlett, M.S., Beveridge, J.R.*, "Recognizing faces with PCA and ICA", Computer Vision and Image Understanding, 91:115-137, 2003.

[3] *Alter O, Brown PO, Botstein D.,* "Singular value decomposition for genome-wide expression data processing and modeling", Proc Natl. Acad. Sci., 97:10101–10106, 2000.

[4] *Craw, I., Costen, N.P., Kato, T., Akamatsu, S.,* "How should we represent faces for automatic recognition?", IEEE Trans. Pat. Anal. Mach. Intel. 21725–736, 1999.

[5] *Grother, P.,* "Software Tools for an Eigenface Implementation" National Institute of Standards and Technology, (2000) http://www.nist.gov/humanid/feret/

[6] *Xiang C.* Face recognition using recursive Fisher linear discriminant / C. Xiang, X.A. Fan, T.H. Lee // Communications, Circuits and Systems. – 2004. – Vol.2. – P. 27-29.

[7] *Conover, W.J.,* Practical Nonparametric Statistics, Wiley, 1980.

[8] *Christel-loic Tisse, Lionel Martin, Lionel Torres, Michel Robert.* Person identification technique using human iris recognition. Proc. of Vision Interface, pp.294-299, 2002.

[9] *Daugman, J.* (2004). How Iris Recognition Works. IEEE Transactions on Circuits and Systems for Video Technology, 14 (1): 21-304.

[10] *Cover, T.M., Thomas, J.A.* (1991). Elements of Information Theory New York:Wiley.

Поступила в редколлегию 10.04.2012

Горбенко Иван Дмитриевич, фото и сведения об авторе см. на с. 190.

Олешко Инна Викторовна, аспирант каф. БИТ ХНУРЭ. Область научных интересов: электронная паспортная система, биометрическая аутентификация

УДК 621. 391:519.2:519.7

Метод оцінки відносної ентропії та порівняльний аналіз джерел біометричної інформації/ І.Д. Горбенко, І.В. Олешко // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 255–261.

У роботі розвивається підхід до вимірювання інформації, що міститься в біометричній характеристиці. Показується, що особливості біометричної інформації можуть бути розраховані за допомогою відносної ентропії. Виконується порівняльний аналіз джерел біометричної інформації.

Ключові слова: біометрична інформація, відносна ентропія, біометрична ідентифікація, матриця коваріації, метод головних компонент, Гауссовий розподіл.

Табл. 1. Іл. 8. Бібліогр.: 10 найм.

UDC 621. 391:519.2:519.7

Method of assessing of relative entropy and comparative analysis of biometric information sources/ I.D. Gorbenko, I.V. Oleshko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 255–261.

This paper develops the approach to the measurement of the information contained in a biometric characteristic. It is shown that the characteristics of biometric data can be calculated using relative entropy. Comparative analysis of biometric information sources was performed.

Keywords: biometric information, relative entropy, biometric identification, covariance matrix, principal components method, Gaussian distribution.

Tab. 1. Fig. 8. Ref.: 10 items.

УДК 621.391:519.2:519.7

АНАЛІЗ ТРЬОХ БІОМЕТРИЧНИХ МЕТОДІВ АВТЕНТИФІКАЦІЇ ОСОБИ

Х.А. БУГАЄНКО, І.Д. ГОРБЕНКО

Вивчаються стандартні механізми автентифікації особи по відбиткам пальців, по райдужній оболочці ока та по геометрії обличчя Vocord Face Control по 2D-зображеню та 3D-моделям, їх можливості, переваги та недоліки. Проводиться аналіз методів автентифікації особи та вносяться пропозиції по їх вдосконаленню.

Ключові слова: автентифікація, метод, особа, біометрична система, обробка зображення, відбиток пальця, райдужна оболонка ока, геометрія обличчя.

ВВЕДЕННЯ

Актуальність розвитку біометричних технологій ідентифікації особи обумовлена збільшенням числа об'єктів і потоків інформації, які необхідно захищати від несанкціонованого доступу, а саме: криміналістика; системи контролю доступу; системи ідентифікації особи; інформаційна безпека (доступ в мережу, вхід на ПК); облік робочого часу та реєстрація відвідувачів; системи голосування, проведення електронних платежів; автентифікація на Web- ресурсах; різні соціальні проекти, де потрібна ідентифікація людей; проекти цивільної ідентифікації (перетин державних кордонів, видача віз на відвідування країни) і т.д.

Предмет дослідження – стандартні механізми автентифікації особи по відбиткам пальців, по райдужній оболочці ока та по геометрії обличчя Vocord FaceControl по 2D-зображеню та 3D-моделям, їх можливості, переваги та недоліки.

Мета роботи – аналіз методів автентифікації особи та пропозиції по їх вдосконаленню.

Метод дослідження – вивчення літератури, обробка та отримання певних результатів.

На відміну від паперових ідентифікаторів (паспорт, водійські права), пароля або персонального ідентифікаційного номера (PIN), біометричні характеристики не можуть бути забуті або загублені, їх важко підробити і практично неможливо змінити.

Одним із засобів забезпечення доступності інформації є автентифікація. Автентифікація - це перевірка приналежності суб'єкту чи об'єкту доступу пред'явленого їм ідентифікатора; перевірка справжності.

В даній статі будуть розглянуті механізми автентифікації особи за рахунок різних біометричних даних та методів.

Ідентифікація на основі біометричних даних - це засіб автоматичного розпізнавання особистості на базі унікальних фізичних або поведінкових параметрів. Ідентифікація виконується за допомогою порівняння отриманих біометричних характеристик і шаблонів, що зберігаються у базі даних.

В даній статі будуть розглянуті механізми автентифікації особи за рахунок різних біометричних даних та методів. Проаналізувавши, зроблені відповідні висновки щодо найефективнішого методу автентифікації особи.

1. ОСНОВНІ ГАЛУЗІ ЗАСТОСУВАННЯ БІОМЕТРИЧНИХ ДАНИХ

Інтеграція Україні до Європи зачіпає не лише економічну і політичну, але і соціальну сферу життя наших громадян. Європейський союз визначив набір протоколів для реалізації EAC (доступу до критичних даних (біометричних даних) в своїх електронних паспортах (ЕП, ePassport)).В Україні і в країнах ЄС ведеться робота по впровадженню біометричних паспортів.

Біометричний паспорт - це документ, що дає право на виїзд за межі країни і в'їзд до іноземних країн. Основною відмінністю електронних документів від існуючих паперових аналогів є те, що в них може бути внесений біометричний набір характеристик, замінити які важче ніж надрукований набір даних, і відповідно набагато важче видавати себе за власника паспорта. Біометричні дані є особливо критичною інформацією, доступ до якої мають отримувати виключно ті системи перевірки, які можуть підтвердити свої повноваження на дані дії. Основними джерелами з питань біометричного паспорту є різноманітні публікації організації цивільної авіації ICAO. Це, в першу чергу, Doc 9303, що складається з кількох частин. Основоположною є частина 1 «Машинозчитуємі паспорти» том 2 «Специфікації на електронні паспорти з засобами біометричної ідентифікації».

Основними російськими стартами по біометричній автентифікації є ГОСТ Р ИСО/МЭК 19794-6 —2006, ГОСТ Р ИСО/МЭК 19794-5 —2006, ГОСТ Р ИСО/МЭК 19794-2 —2005, ГОСТ Р ИСО/МЭК 19794-4 —2006, ГОСТ Р ИСО/МЭК 19785-1 —2008, ГОСТ Р ИСО/МЭК 19794-7 —2009, ГОСТ Р ИСО/МЭК 19795-2 —2006.

Ідентифікація на основі біометричних даних - це засіб автоматичного розпізнавання особистості на базі унікальних фізичних або поведінкових параметрів. Ідентифікація виконується за допомогою порівняння отриманих біометричних характеристик і шаблонів, що зберігаються у базі данних. В залежності від характеристик, які при цьому використовуються.

Біометричні системи поділяються на статичні та динамічні. До методів першої групи відносяться такі види автентифікації: за відбитком пальця, за формою долоні, по розташуванню вен на лицьовій стороні долоні, по сітківці ока, по райдужній оболонці ока, за формою обличчя, по термограмі обличчя, по ДНК, інші методи. До динамічних методів відносять автентифікацію по рукописному почерку, по клавіатурному почерку, по голосу, інші методи. В даний час активно використовуються такі біометричні ознаки, як відбитки пальців, геометрична форма кисті руки, геометрія обличчя, особливості голосу, райдужна оболонка ока, рукописний підпис.

2. АВТЕНТИФІКАЦІЯ ОСІБ ПО ВІДБИТКАМ ПАЛЬЦІВ

Серед всіх біометричних технологій ідентифікація за відбитками пальців є найстарішим і найпоширенішим методом, який успішно застосовується в багатьох областях. У кожної людини свої унікальні і незмінні відбитки пальців. Задача ідентифікації особистості по відбитку пальця вирішується шляхом зіставлення ідентифікованого відбитка з еталонними [1]. Відбитки вважаються ідентичними, якщо коефіцієнт відповідності становить 65% [2] і вище (цей поріг можна змінити).

Відбитки пальців можуть відрізняти один від одного поворотом, зміщенням, зміною масштабу і площею дотику. Пропонований підхід розроблений для порівняння відбитків пальців однакової розмірності і складається з трьох основних етапів [6]:

1. Обробка вихідного зображення (дивись рис. 1).

Рис. 1. Вихідне зображення

1.1. Обчислення орієнтації ліній;
1.2. Поліпшення якості ліній (дивись рис. 2);

Рис. 2. Зображення після покращення якості

1.3 Бінаризація зображення;
1.4 Стоншення ліній зображення (дивись рис. 3).

Рис. 3. Зображення після стоншення ліній

2. Виділення мінуцій (див. рис. 4).

Рис. 4. Визначення мінуцій

3. Зіставлення мінуцій відбитків пальців.
3.1 Знаходження центру;
3.2 Переміщення;
3.3 Поворот;
3.4 Зміна масштабу (не розглядається).

По даному методу можна виділити основні недоліки та переваги, що показано в табл. 1.

Таблиця 1

Переваги та недоліки методу

Переваги методу	Недоліки методу
• висока достовірність – статистичні показники методу вище показників способів ідентифікації по обличчю, голосу, розпису; • низька вартість обладнання; • достатньо проста процедура сканування відбитка.	• папілярний узор відбитка пальця дуже легко можна пошкодити дрібними подряпинами, порізами; • недостатня захищеність від підробки, викликана широким поширенням методу; • залежність від чистоти пальця; • для сухої шкіри якість розпізнавання нижче.

3. АВТЕНТИФІКАЦІЯ ОСІБ ПО РАЙДУЖНІЙ ОБОЛОЧЦІ ОКА

1. Виявлення великого центру зіниці. *X*-координата — це центр зіниці розраховується за формулою:

$$x_0 = x_l + x_r / 2.$$

Можемо отримати верхні та нижні координати краю зіниці в місці розташування:

$$(x_p, y_u), (x_p, y_b).$$

У-координата центру зіниці розраховується за формулою:

$$y_0 = y_u + y_b / 2.$$

В результаті отримаємо великий центр зіниці, який знаходиться по координатам (x_0, y_0).

2. Вирівнювання контуру границі зіниці. Можна записати суму помилок як:

$$E^2 = \sum(x_i^2 + y_i^2 + cx_i + dy_i + e)^2. \qquad (1)$$

Якщо ми беремо приватні похідні з рівняння (1) щодо коефіцієнти *c*, *d* та *e*, *i* – набір кожен до нуля, отримуємо три рівняння та координати x_i, y_i. Ми отримаємо центр зіниці I (I_x, I_y) і радіус R_i через співвідношення вищезгаданих точок краю зіниці. Так само знаходимо центр зовнішньої райдужної оболонки O (O_x, O_y) і радіус R_0.

3. Якісна оцінка зображення райдужної оболонки. Практично, якість деяких зображень райдужної оболонки настільки жахлива, що помилка, 100-відсотково буде існувати. Щоб уникнути цієї проблеми, необхідно оцінити якість райдужної оболонки зображення. Тільки, коли якість відповідає нашому запиту, ми можемо використовувати це зображення. Головні проблеми поганої якості зображення райдужної оболонки — розмиття, викликане поганим фокусуванням, часом та іншими факторами.

В табл. 2 приведені можливі переваги та недоліки даного методу автентифікації.

Таблиця 2

Переваги та недоліки методу

Переваги методу	Недоліки методу
• статистична надійність алгоритму; • захоплення зображення проводиться на відстані від декількох см до декількох метрів, фізичний контакт людини з пристроєм не відбувається; • райдужна оболонка захищена від пошкоджень,тому не змінюється в часі.	• ціна системи, вище ціни системи, заснованої на розпізнаванні пальця або на розпізнаванні особи; • низька доступність готових рішень.

4. АВТЕНТИФІКАЦІЯ ПО ГЕОМЕТРІЇ ОБЛИЧЧЯ VOCORD FACE CONTROL ПО 2D-ЗОБРАЖЕНЮ ТА 3D-МОДЕЛЯМ

VOCORD FaceControl – система автоматичного некооперативного виявлення, простежування і виділення зображень облич, формування векторів ознак по 2D-зображеню і 3D-моделям за рахунок біометричної ідентифікації людини.

А також можна виділити області застосування поданих двох методів, для забезпечення безпеки в місцях масового скупчення людей:

– місця масового перебування людей: площі, стадіони;

– транспортні вузли: аеропорти, вокзали, автостанції;

– додаткові кошти ідентифікації особистості;

– прикордонні паспортно-візові контрольні пункти;

– прохідні і контрольно-пропускні пункти;

– громадські установи.

Для біометричного методу ідентифікації особи по 2D – розпізнаванню розглянемо всі ймовірні переваги та заодно й недоліки, які містяться в даному методі (дивитись табл. 3-4) [5].

Таблиця 3

Переваги та недоліки методу
(2D – розпізнавання особи)

Переваги методу	Недоліки методу
- не потрібне дороге обладнання; - при відповідному обладнанні можливість розпізнавання на значних відстанях від камери.	- низька статистична достовірність; - пред'являються вимоги до освітлення; - неприйнятність будь-яких зовнішніх перешкод; - не враховують можливі зміни міміки обличчя, вираз повинен бути нейтральним.

Допуск по 2D-зображенню зручний і дешевий, але має обмежену сферу застосування через погані статистичних показників. З формули:

$$\mathrm{FAR} * N2 \approx 1 => N = (1/\mathrm{FAR})1/2. \qquad (2)$$

З формули (2) отримуємо $N \approx 30$ - чисельність персоналу організації, при якій ідентифікація співробітника відбувається досить стабільно. Для біометричного методу ідентифікації особи по 3D – моделі розпізнаванню розглянемо всі ймовірні переваги та заодно й недоліки, які містяться в даному методі.

Таблиця 4

Переваги та недоліки методу
(3D – розпізнавання особи)

Переваги методу	Недоліки методу
• висока достовірність розпізнавання –більше інформації, чим має звичайний знімок; • стійкість розпізнавання до відхилення ракурсу особи від фронтального; • стійкість розпізнавання до неоднорідності освітлення; • відсутність необхідності контактувати з пристроєм; • низька чутливість до зовнішніх факторів.	• має обмежену сферу застосування із-за поганих статистичних показників; • дороге обладнання; • зміна міміки обличчя і перешкоди на обличчі погіршують статистичну надійність методу.

В табл. 5 зведені результати розглянутих двох методів, що базуються на автентифікації особи по геометрії обличчя.

Системи розпізнавання по 3D – зображенню особи дуже специфічні. Вони можуть знадобитися у випадках, коли розпізнавання вимагає відсутності фізичного контакту, але поставити

систему контролю по райдужній оболонці неможливо.

Таблиця 5

Аналіз двох методів

	Стійкість до підробки	Стійкість до довкілля	Простота використання	Вартість	Швидкість	Стабільність біометричної ознаки в часі
Обличчя 2D	4	6	6	10	10	8
Обличчя 3D	9	8	10	5	7	10

Повні дані про FRR і FAR для алгоритмів по 3D – зображенню на сайтах виробників відкрито не наведено. Але для кращих моделей фірми Bioscript (3D EnrolCam, 3D FastPass), які працюють за методом проектування шаблону при FAR = 0.0047% FRR становить 0.103%. Вважається, що статистична надійність методу порівнянна з надійністю методу ідентифікації за відбитками пальців.

ВИСНОВОК

Ця робота присвячена актуальному питанню – ідентифікації особи на основі біометричних даних. Ця методика має ряд переваг: використання тільки фізичних параметрів людини; неможливість підробки біометричних даних.

Ймовірність того, що відбитки пальців у двох людей співпадуть - один до 220 мільйонів. Тобто $1/220000000 = 4{,}54*10^{-9}$.

З райдужної оболонки 11-міліметрового діаметру сучасні алгоритми обробки і аналізу інформації дозволяють отримати в середньому 3,4 біт інформації на 1 мм^2 площі [3, 4]. Щільність отримуваної інформації така що райдужна оболонка має 266 унікальних точок ідентифікації в порівнянні з 10-60 точками для інших біометричних методів [4].

Хоча біометрія представляє собою інтерес як джерело ентропії, дослідження вказують на те, що багато біометричних методів, можливо, не фактично, але пропонують достатню невпевненість з цією метою.

Табл. 6 та 7 ілюструють звітність отриманих результатів по обраним трьом методам автентифікації.

По показнику FAR (false acceptance rate) найбільш надійним буде метод по 2D / 3D моделям, тому що ймовірність хибного співпадання біометричних характеристик двох осіб найменша в порівняні з останніми двома методами.

По показнику FRR (false rejection rate) найбільш надійним буде метод по райдужній оболонці ока, тому що ймовірність відмови в доступі особи, що має право доступу найменша, а найбільш гіршим по цьому показнику являється метод по відбиткам пальців.

Таблиця 6

Звітність отриманих результатів

Назва методу	FAR	FRR	Ймовірність співпадання	Швидкість обробки	Попит на ринку
По відбиткам пальців	0,01%	10%	$4{,}54* 10^{-9}$	$2{,}5*10^{-4}$, с	39%
По райдужній оболонці ока	0,01%	0,05%	10^{-78}	0,3 - 0,5, с	6% - 9%
По 2D / 3D моделям	0,0047%	0,103%	$3{,}26* 10^{-7}$	1-2, с	13% -18%

Таблиця 7

Звітність отриманих результатів

Назва методу	Стійкість до підробки	Стійкість до довкілля	Простота використання	Вартість
По відбиткам пальців	6	10	9	10
По райдужній оболонці ока	10	9	8	7
По 2D / 3D моделям	4/9	6/8	6/10	10/5

Оцінюючи по ймовірності співпадання однакових зображень маємо, що найбільш кращим являється метод по райдужній оболонці ока, так як цей метод має мінімальну ймовірність в порівнянні з двома іншими, а гіршим буде – по 2D / 3D моделям.

Використовуючи дані щодо швидкості обробки одного зображення, то кращим являється метод по відбиткам пальців, а потім йдуть по райдужній оболонці ока та по 2D / 3D моделям.

Оцінюючи методи по стійкості до підробки можна сказати, що під цей параметр підходить як метод по райдужній оболонці ока так і метод по 2D / 3D моделям. По стійкості до довкілля найкращими будуть методи по відбиткам пальців та по райдужній оболонці ока, тому що стійкість роботи системи за різних зовнішніх умов, таких як зміна освітлення або температури приміщення, не залежить.

По показнику складність використання, тобто наскільки складно скористатися біометричним приладом, взагалі то під цей показник попадають усі три методи. Беручи до уваги вартість приладів для ідентифікації особи, то можна сказати, що на даний момент найбільш дорогим буде придбання приладу для ідентифікації особи по 2D / 3D моделям. А найбільш дешевшим являються прилади для ідентифікації особи по відбиткам пальців.

По показнику – попит на ринку [5], то найбільш використовуваним являється ідентифікація особи по відбиткам пальців, яка в свою чергу

займає майже половину світового ринку, потім йде геометрія обличчя (2D / 3D - моделі) та на останньому місці опинився метод по райдужній оболочці ока.

Для райдужної оболонки ока можна збільшити точність системи практично квадратично, без втрат для часу, якщо ускладнити систему, зробивши її на два ока. Для відбитків пальців - шляхом комбінування декількох пальців, шляхом комбінування двох рук, але таке поліпшення можливо тільки при збільшенні часу, що витрачається при роботі з людиною.

Узагальнивши результати для методів, можна сказати, що для середніх і великих об'єктів а так само для об'єктів з максимальною вимогою у безпеці слід використовувати райдужну оболонку в якості біометричного доступу.

Для об'єктів з кількістю персоналу до декількох сотень чоловік оптимальними буде доступ по відбитках пальців. Системи розпізнавання по 3D зображенню особи дуже специфічні. Вони можуть знадобитися у випадках коли розпізнавання вимагає відсутності фізичного контакту, але поставити систему контролю по райдужній оболонці неможливо. Наприклад, при необхідності ідентифікації людини без її участі, прихованою камерою, або камерою зовнішнього виявлення, але можливо це лише при малій кількості суб'єктів у базі і невеликому потоці людей, що знімаються камерою.

Література

[1] *Anil K.* Jain Handbook of Biometrics. Springer Science+Business Media, 2008.

[2] *В. Задорожный,* Идентификация по отпечаткам пальцев. PC Magazine, - 2004, - т.1 - №2.

[3] *Daugman J G.* Biometric personal identification system based on iris analysis - [P].US Patent 5291560, 1994.

[4] *Wildes R P.* Iris recognition, An Emerging Biometric Technology. //.Processing of the IEEE – с.185, 1997.

[5] *Chang, K. I.* Multiple nose region matching for 3D face recognition under varying facial expression. / K. I. Chang// IEEE Transactions on Pattern Analysis and Machine Intelligence– №28(10), October 2006. –1695–1700 с.

[6] *H. Lin, A. Jian, S. Pankanti, R. Bolle,* Fingerprint enhancement. //Applications of Computer Vision, 1996.

Надійшла до редколегії 18.04.2012

Горбенко Іван Дмитрович, фото та відомості про автора див. на стор. 190.

Бугаєнко Христина Андріївна, студентка кафедри безпеки інформаційних технологій Харківського національного університету радіоелектроніки. Наукові інтереси: біометричні методи, їх ефективність, попит на світовому ринку та можливі переваги та недоліки.

УДК 621.391:519.2:519.7

Анализ трех биометрических методов аутентификации личности / К.А. Бугаенко, И.Д. Горбенко // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 262–266.

Изучаются стандартные механизмы аутентификации личности по отпечаткам пальцев, по радужной оболочке глаза и по геометрии лица Vocord Face Control по 2D-изображению и 3D-моделям, их возможности, преимущества и недостатки. Проводится анализ методов аутентификации личности и вносятся предложения по их усовершенствованию.

Ключевые слова: аутентификация, метод, личность, биометрическая система, обработка изображения, отпечаток пальца, радужная оболочка глаза, геометрия лица.

Табл. 7. Ил. 4. Библиогр.: 6 назв.

UDC 621.391:519.2:519.7

Analysis of three biometrical methods of person authentication/ K.A. Bugaenko, I.D. Gorbenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 262–266.

The paper studies standard mechanisms for authenticating a person by fingerprints, by eye iris and the geometry of the face Vocord Face Control of 2D-images and 3D-models, their features, advantages and disadvantages. Besides, analysis of methods of person authentication is done and proposals to perfect them are introduced.

Keywords: authentication, method, person, biometric system, image processing, fingerprint, iris, face geometry.

Tab. 7. Fig. 4. Ref.: 6 items.

УДК 343.982.3, 004.89, 004.032.26

АНАЛІЗ БІОМЕТРИЧНИХ ІНТЕЛЕКТУАЛЬНИХ МЕТОДІВ АВТЕНТИФІКАЦІЇ ТА ІДЕНТИФІКАЦІЇ ОСОБИ ЗА ВІДБИТКАМИ ПАЛЬЦІВ ТА ЗА ГОЛОСОМ ДЛЯ ЗАХИСТУ ВІД НЕСАНКЦІОНОВАНОГО ДОСТУПУ

П.О. ФІЛОНЕНКО, Є.І. БАРСУКОВ, О.А. ВИНОКУРОВА

У статі наводяться результати аналізу проблемних питань систем автентифікації осіб. Основною метою аналізу є визначення перспективної моделі біометричної автентифікації та ідентифікації особи для впровадження та застосування в Україні. До основних задач відносяться аналіз існуючих методів біометричної автентифікації та ідентифікації особи за відбитками пальців та за голосом для підвищення стійкості від загроз в сфері інформаційної безпеки. Розглянуто метод виявлення локальних особливостей біометричних образів за допомогою гібридних штучних нейронних мереж.

Ключові слова: біометрія, автентифікація, гібридні інтелектуальні методи, штучні нейронні мережі.

ВСТУП

Існуючі на сьогоднішній день традиційні методи верифікації особи, які засновані на зберіганні певних ключових даних або запам'ятовуванні паролів, не завжди є надійними і зручними, так як є досить висока ймовірність того що ці дані можуть бути загублені або забуті. Для підвищення надійності проходження процедури автентифікації природним кроком стало використання в системах безпеки біометричних технологій.

На сьогоднішній день біометричні системи використовуються за двома напрямками: для контролю фізичного доступу та доступу до інформації. Такі рішення реалізуються на різних споживчих рівнях: приватному, корпоративному, державному, міждержавному.

Для всіх біометричних методів верифікації характерно величезна відмінність між показниками ефективності в лабораторних умовах, які повідомляються розробниками, і результатами тестування незалежними організаціями. Деякі з перспективних методів приведені у табл. 2.1 [1].

Таблиця 1

Незалежні оцінки систем біометрії

Тип біометрії	Тестові організації	Відсоток хибних відмов	Відсоток хибних пропусків
Відбитки пальців (4 пальця)	Fingerprint Verification Competition (2004)	2 %	2 %
Параметри голосу	The National Institute of Standards and Technology (NIST), (2004)	5-10 %	2–5 %

1. ІЄРАРХІЯ БІОМЕТРИЧНИХ СТАНДАРТІВ

Розглянемо стандарти, зазначені в ієрархії, і організації, що займаються їх створенням. Виняток зробимо лише для стандартів щодо розрахунку продуктивності та інших технічних характеристик біометричних систем, тому що вони зараз самі "сирі" і багато в чому безпосередньо залежать від стану та затвердження стандартів інших груп (рис. 1).

У 2001 р. в США при Міжнародному комітеті з стандартам в інформаційних технологіях (International Committee for IT Standards, INCITS) у листопаді того ж року був створений технічний комітет M1, основним завданням якого стала прискорена розробка стандартів з біометрії для використання в США і в міжнародних стандартах.

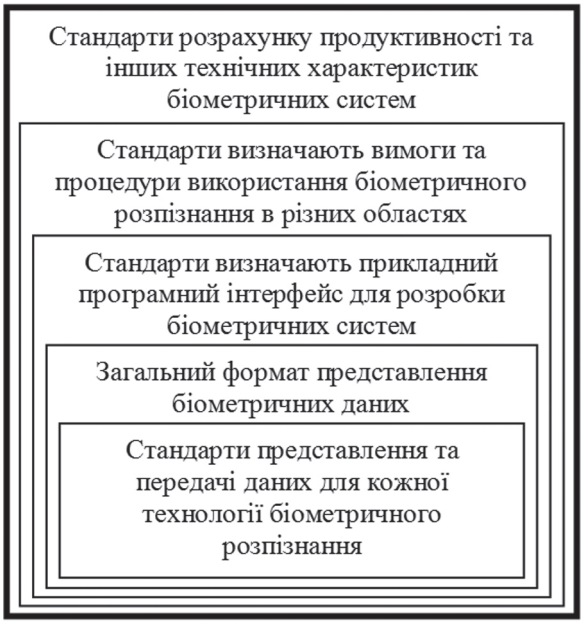


Рис. 1. Ієрархія стандартів з біометрії

На міжнародному рівні цими завданнями займається підкомітет SC37 (Subcommittee 37) об'єднаного технічного комітету з інформаційних технологій JTC 1 Міжнародної організації стандартизації ISO (International Organization for Standardization) та Міжнародної електротехнічної комісії (International Electrotechnical Commission, IEC), створений у червні 2002 р. [2].

У нього входить близько 20 країн, в тому числі і Росія. У Росії створено підкомітет N7 по

біометрії в рамках Технічного комітету Держстандарту ТК355 по автоматичної ідентифікації, який діє у складі SC37 і займається адаптацією і випуском російських стандартів з біометрії. Група М1 входить до SC37 як представник США і технічна консультаційна група (Technical Advisory Group) (рис. 2).

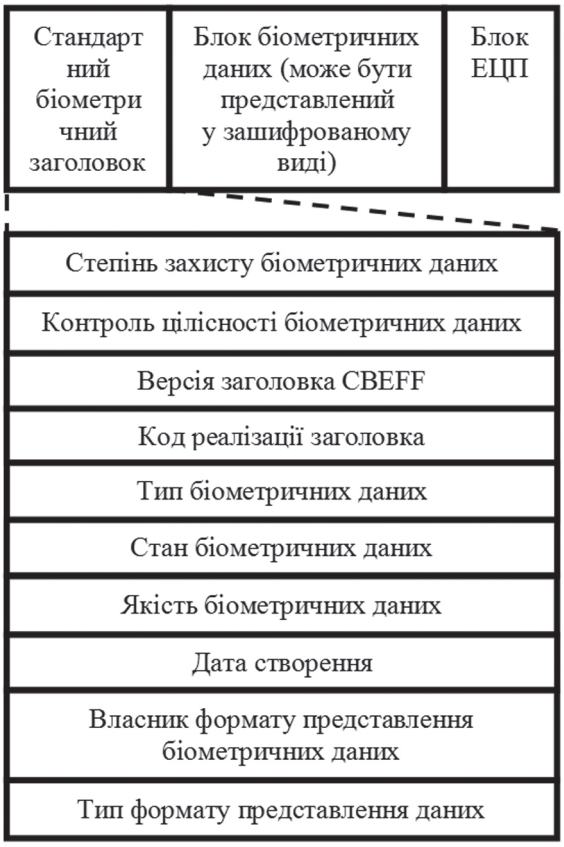


Рис. 2. Склад Common Biometric Exchange File Format

Напрямки діяльності М1 і SC37 аналогічні, тому зупинимося на описі робіт М1, так як вони почалися раніше і деякі з них лягли в основу біометричних стандартів JTC1 SC 37 ISO / IEC.

Використання біометрії в сфері фінансових послуг - стандарт X9.84.

З стандартів, які визначають вимоги щодо використання біометрії в різних промислових галузях, прийнятий стандарт "ANSI X9.84-2000. Biometrics Management and Security for the Financial Services Industry" [3] Американського національного інституту стандартів, розроблений робочою групою X9.F4 акредитованого ANSI комітету стандартів X9. Згодом вийшла його оновлена версія X9.84-2003.

X9.84 визначає мінімальні вимоги безпеки при побудові біометричних систем для сфери фінансових послуг, а також механізми і правила криптографічного захисту процесів одержання, обробки і зберігання біометричних даних.

Зокрема, в стандарті викладені вимоги за такими темами:

– управління біометричними даними та їх захист під час життєвого циклу;

– використання біометричної технології для ідентифікації і автентифікації співробітників і клієнтів банків;

– застосування біометричної технології в системах контролю і управління доступом;

– інкапсуляція біометричних даних;

– технологія захищеної передачі біометричних даних.

Виділимо основні вимоги щодо безпеки для біометричних систем в X9.84.

– біометрична система повинна запобігати можливість обробки біометричних даних, що надійшли до системи з неавторизованого зчитувального біометричного пристрою.

– біометрична система повинна бути побудована так, щоб біометричні дані могли вступити до неї тільки через авторизовані інтерфейси з використанням прийнятих процедур.

– у біометричну систему повинні бути вбудовані механізми захисту для виявлення і запобігання використання штучних біометричних характеристик (наприклад, муляжів).

– там, де це необхідно, в біометричну систему повинні вбудовуватися механізми захисту для запобігання витоку або втрати біометричних даних.

– біометрична система повинна обмежувати доступ до шаблонів, тобто запобігати можливість: реконструкції бази шаблонів за допомогою перехоплених біометричних даних; - обробки запитів на верифікацію в обхід бази шаблонів.

Ідеологія, з якої виходили творці X9.84, спиралася на те, що біометричні характеристики людини не абсолютно конфіденційні і не є прихованими даними (голос можна записати на плівку, обличчя і радужку очей сфотографувати, відбиток пальця зняти з предмета і т. д.) . За певних умов біометричні характеристики можна підробити. Тому весь життєвий цикл всередині біометричної системи по X9.84 повинен бути захищений, скануючи засоби авторизовані в системі, всі дані зашифровані, а сама система повинна вміти відрізняти реальні біометричні характеристики людини від їх підробок.

2. АНАЛІЗ МЕТОДІВ БІОМЕТРИЧНОЇ АВТЕНТИФІКАЦІЇ З ВИКОРИСТАННЯМ ВІДБИТКІВ ПАЛЬЦІВ

Для обчислення біометричного образу за відбитками пальців було проаналізовано існуючий запатентований алгоритми Precise BioMatch та Міжнародний стандарт ISO/IEC 19794 – 2:2005.

Алгоритм Precise BioMatch [4] використовує як переваги традиційних методів виділення ключових точок, так і передові алгоритми порівнювання візерунків. Такий подвійний підхід дозволяє отримати максимальну кількість інформації з відбитку для подальшого якісного аналізу та гарантування вірною автентифікації. Precise BioMatch створена не тільки для алгоритмів автентифікації особистості у великій базі даних (як,

наприклад, алгоритм AFIS – автоматизовані системи ідентифікації відбитків пальців), але і для найкращого підтвердження особи в логічному і фізичному доступі.

Алгоритм Precise BioMatch не прив'язаний до конкретного типу датчика (сканера відбитків). Отже, користувач може зареєструвати відбитки на одному типі датчика, а проходити перевірку на іншому. Це надзвичайно важливо у випадках, коли біометричне розпізнавання відбитка пальця використовується на дуже великій, неконтрольованій відстані. Але як зазначалось це є запатентований метод і для його використання потрібна згода власників.

Міжнародний стандарт ISO/IEC 19794 – 2:2005 встановлює структуру і формат блоку даних по контрольних точках зображення відбитка пальця .

Стандарт поширюється на широкий діапазон прикладних областей, що використовують автоматизоване розпізнавання відбитку пальця. Стандарт містить терміни та визначення, опис правил визначення контрольних точок, формати даних, у тому числі для використання в ідентифікаційних картах [5].

Цей стандарт встановлює правила визначення розташування контрольних точок на відбитку пальця. Для забезпечення взаємодії між різними біометричними системами на підставі розпізнавання відбитків пальців і порівняння індивідуальних і попередньо зареєстрованих записів відбитків пальця необхідно гарантувати сумісність різних методів отримання контрольних точок. Працює з методів досягається дотриманням правил вилучення контрольних точок відбитка пальця, правил запису форматів і форматів ідентифікаційних карт, які є загальними для біометричних систем, і передбачає можливість введення розширених біометричних даних, сумісних з конкретним обладнанням.

Використання подання відбитка пальця з допомогою характерних ознак спирається на загальноприйняту практику. Контрольними точками називають точки, розташовані на зображенні відбитка пальця в місцях закінчення відбитків гребенів або в місцях біфуркації гребенів. Опис зображення відбитка пальця про терміни розташування і орієнтації контрольних точок закінчення і біфуркацій гребенів дозволяє гарантовано визначити, чи є два зображення відбитками одного і того ж пальця. Цей стандарт встановлює правила визначення та кодування розташування і орієнтації контрольних точок.

Існує два основних типи контрольних точок: точка закінчення основи гребеня і точка біфуркації основи гребеня (або точка розгалуження). Крім зазначених типів у відбитках пальців рідше зустрічаються й інші типи інформативних точок, що мають більш складні визначення. Більш складні типи контрольних точок зазвичай є комбінаціями основних типів, зазначених вище. Деякі кінь рольні точки не є ні точками закінчення гребенів, ні точками біфуркації. Подібні точки відносять до додаткового типу «інша контрольна крапка». Тип «інша контрольна точка» не слід використовувати для контрольних точок закінчення гребеня або біфуркації гребеня. Таким чином, цей Стандарт встановлює наступні типи контрольних точок:

– закінчення гребеня (точка біфуркації основи западин);
– біфуркація гребеня;
– інша контрольна крапка.

У залежності від методу визначення положення точки допускається визначати контрольну точку закінчення гребеня як точку біфуркації западини. Вид методу кодування контрольних точок за допомогою точки закінчення гребеня або точки біфуркації западини повинен бути вказана в полі «Тип формату» біометричного інформаційного шаблону.

Розташування контрольної точки визначають за її горизонтальному і вертикальному положенням. Пошук контрольних точок слід проводити на засадах гребенів або западин, витягнутих з цифрового зображення відбитка пальця. Основу гребеня обчислюють поетапним зменшенням зображення гребеня до лінії шириною в один елемент зображення. Основу западини обчислюють поетапним зменшенням площі западини до лінії шириною в один елемент зображення.

Використання інших методів виявлення контрольних точок допускається тільки у випадку, якщо їх результати відповідають результатам методу стоншення, тобто, якщо значення розташувань і орієнтацій контрольних точок, отримані іншим методом, еквівалентні значенням розташуванні й орієнтації контрольних точок, отриманим методом стоншення.

Обчислення координат контрольних точок слід проводити в декартовій системі координат X-Y. Початок системи координат зображення відбитка пальця має розташовуватися в лівому верхньому кутку вихідного зображення. Вісь X за загальноприйнятим у цифровій обробці зображень допущенню повинна бути спрямована зліва направо (позитивний напрямок), вісь Y повинна бути спрямована вниз (позитивний напрямок). У системі координат зображення пальця вісь X повинна бути направлена справа наліво відповідно до рис. 3. Всі значення координат X і Y повинні бути невід'ємними.

Рис. 3. Система координат

Координати X і Y контрольних точок слід визначати з кроком, рівним одному елементу зображення, і з просторовим дозволом, наведеним у полях «Дозвіл по осі X» і «Дозвіл по осі Y». Дозволи зображення по осі X і Y визначають окремо.

У форматі запису контрольних точок відбитка пальця дозвіл системи координат має бути записано в заголовку запису.

Цей стандарт встановлює такі правила визначення та запису значень кутів. Кут орієнтації контрольних точок вимірюють від горизонтальної осі проти годинникової стрілки.

У форматах запису кут орієнтації контрольних точок квантується з кроком квантування рівним куту 1,40625° (360/256], на один молодший біт. Кодування кута у форматах для використання в ідентифікаційних картах залежить від того, що використається формат нормального або компактного розміру.

Орієнтація контрольної точки закінчення гребеня, визначеної через точку біфуркації основи западин.

Контрольна точка закінчення гребеня, визначена через точку біфуркації основи западин, відповідає трьом лініям западин, що зустрічаються в одній точці. При цьому дві западини утворюють гострий кут, а дотична до третьої западині, протилежної лінії гребеня, визначає напрямок біфуркації западини. Напрямок контрольної точки слід вимірювати як значення кута між зазначеної дотичній та горизонтальною віссю, орієнтованої вправо (рис. 4).

Орієнтація контрольної точки біфуркації гребеня, визначеної через точку біфуркації основи гребеня.

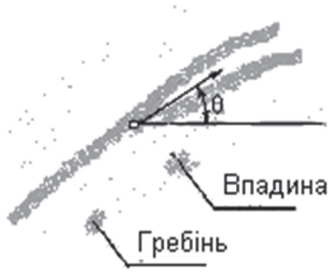


Рис. 4. Розташування і орієнтація контрольної точки закінчення гребеня певної через точку біфуркацій основи западин

Контрольна точка біфуркації гребеня, визначена через точку біфуркації основи гребеня, яка відповідає трьом лініям гребенів, зустрічається в одній точці. При цьому два гребені утворюють гострий кут, а дотична до третього гребеню, протилежному западині, визначає напрямок біфуркації гребня. Напрямок контрольної точки слід вимірювати як значення кута між вказаною дотичної та горизонтальною віссю, орієнтованої вправо.

Напрямок контрольної точки завершення основи гребенів слід вимірювати як значення кута, утвореного дотичній до закінчення гребеня і горизонтальною віссю, орієнтованої вправо. Точку закінчення основи гребенів використовують тільки в одному з двох варіантів форматів, використовуваних в ідентифікаційних картах, в інших варіантах формату використовують точки закінчення гребеня і біфуркації гребеня.

Ядро і дельта є інформативними точками відбитка пальця. Відбиток пальця може не мати або мати одну або більше дельт, а також мати одну чи більше ядер. Цей стандарт встановлює наступні правил визначення розташування та орієнтації ядра і дельти.

Розташування ядра: якщо на зображенні відбитка пальця присутній контрольна точка закінчення гребеня поблизу самого внутрішнього загину гребеня, то розташування ядра визначають за розташуванням контрольної точки закінчення гребеня, найбільш близькою до гребньовій лінії, що має максимальну кривизну. Якщо ядро має вигляд перевернутої букви «U» без найближчих контрольних точок закінчення гребеня, то розташування ядра визначають за розташуванням відповідної контрольної точки закінчення впадини.

Орієнтація ядра: якщо ядро характеризується вираженим напрямком, то значення кута цього напрямку повинно бути записане в полі «Орієнтація ядра», що входить в структуру формату запису контрольних точок. Орієнтацію ядра визначають за значенням кута дотичній до гребневих лініях, розташованим поблизу ядра: напрям дотичній слід визначати з відкритої сторони опуклого гребеня.

Розташування дельти: для визначення розташування дельти необхідно встановити три додаткові точки, кожна з яких розташована між двома сусідніми гребенями в області розбіжності гребенів: тобто в області, в якій паралельні або майже паралельні гребневі лінії розходяться при наближенні до дельті. Розташування дельти визначають як центр мас цих трьох точок.

Орієнтація дельти: для всіх розбіжностей гребеневих ліній визначають кут нахилу дотичної до гребенів в точці, розташованої до розбіжності ліній гребенів. Кут вимірюють від точки розбіжності паралельних гребенів у напрямку від дельти. Розташування ядра і дельти наведено на рис. 5.

Рис. 5. Приклад розташування ядра і дельти

Після створення цифрового біометричного образу за допомогою сканерів, зберігання їх

у базі суттєво зменшує стійкість від атаки зловмисника. Отже образ зберігається в системі до того часу поки не будуть зроблені наступні дії, а саме обчислення біометричних параметрів з пред'явленого образу.

3. АНАЛІЗ МЕТОДІВ БІОМЕТРИЧНОЇ АВТЕНТИФІКАЦІЇ ЗА ГОЛОСОМ

Системи голосової біометрії не вимагають дорогої апаратної підтримки, універсальність полягає в можливості використання як при безпосередньому контакті з реєструючої апаратурою, так і при віддаленому доступі, наприклад, по каналах телефонних дротових або мобільних ліній. Це дає можливість легко адаптувати системи автентифікації на основі голосової біометрії до різних умов використання і сферам застосування. Тому голосова біометрія є перспективним методом верифікації особистості як з точки зору надійності, так і з точки зору широти областей застосування.

Для обчислення біометричного образу за голосом були проаналізовані декілька перспективних методів розроблених провідними компаніями.

Voice Key Service – система голосової біометричної автентифікації, розроблена російською компанією «Центр мовних технологій» (ЦРТ). Технологія Voice Key використовує унікальні характеристики фізіологічної будови мовного тракту кожної людини. В її основі лежить запатентований компанією ЦРТ алгоритм, який використовує спектрально-формантний метод виділення і порівняння біометричних ознак.

Переваги цієї системи полягають у тому, що дана система має:

– два рівня захисту (порівняння біометричних даних + перевірка пароля);

– верифікація у телефонному каналі;

– можливість працювати в зашумлених умовах;

– незалежність від національної мови або діалекту.

Недоліком є те, що система не володіє можливістю встановлення параметрів для кожної програми.

SPIRIT SV-система – система автентифікації, розроблена російською компанією SPIRIT Corp. Ця система здатні працювати в різних додатках: від автентифікації диктора для локальних систем безпеки до віддаленої автентифікації по телефону, що може бути застосовано, наприклад, для банківських служб та електронної комерції. Конкретне рішення може бути зроблено SPIRIT Corp., Включаючи портінг системи на задану платформу та забезпечення телекомунікаційної підтримки.

Переваги даної системи:

– можливість автентифікації в телефонному каналі;

– можливість працювати в зашумлених умовах;

– незалежність відмов і словників;

– здатна працювати у текстозалежному режимі і в режимі підказок.

Недоліком даної системи є те, що для надійної роботи вимагає обмеження на 10-15 користувачів, що не підходить для використання в умовах більшої чисельності користувачів системи доступу, відсутня можливість додаткової автентифікації (перевірки введеного немовного пароля, наприклад, з клавіатури) для збільшення рівня надійності , система не володіє можливістю встановлення параметрів для кожної програми [6].

Speech Secure – система ідентифікації голосу, розроблена американською компанії Nuance Technology. Спочатку в процесі реєстрації, система за спеціальними алгоритмами, створює модель голосу, використовуючи унікальні характеристики голосу того, хто телефонує. Система зберігає моделі голосу (опис структури голосу і особливостей голосового тракту) як частину профілю абонента. Під час автентифікації (ідентифікації) ці моделі використовуються для визначення ступеня відповідності голосу того, хто телефонує голосам записаних раніше людей. На основі цієї інформації система приймає рішення щодо проведення операції. Система доступна через веб-інтерфейс.

Повна версія включає: машину автентифікації, біометричне додаток ідентифікує людину за унікальною голосової моделі, сервер, веб-сервіси для використання з будь голосової платформою з управлінням базою даних голосових моделей.

Переваги:

– легко інтегрується в систему будь-якої архітектури;

– можливість працювати в зашумлених умовах;

– зменшує ймовірність фальсифікацій і шахрайства при використанні бази даних підозрілих голосів і перемиканні підозрілих абонентів на службу безпеки.

Недолік – володіє надлишком функцій, внаслідок чого має складну настройку.

Більшість сучасних систем зосереджують зусилля на добуванні частотної характеристики мовного тракту людини, відкидаючи при цьому характеристики сигналу збудження [7]. Для відділення сигналу збудження від сигналу мовного тракту вдаються до кепстрального аналізу. Схематично цей метод представлений на рис. 6.

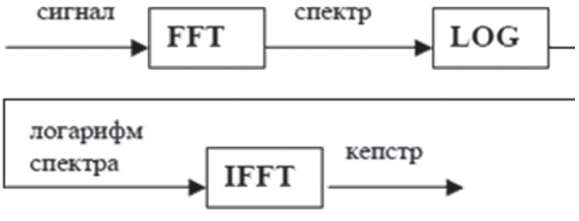


Рис. 6. Загальна схема кепстрального аналізу сигналу (FFT – блок швидкого перетворення Фур'є сигналу, LOG – блок логарифмування спектру, IFFT – блок зворотного швидкого перетворення Фур'є)

Одним із потужних методів, заснованим на кепстральному аналізі сигналу, відноситься метод кепстральних коефіцієнтів лінійного передбачення (LPCC). Перші два етапи цифрової обробки полягають у попередньому посиленні(pre-emphasis) та сегментації на фрейми.

На першому етапі до сигналу застосовується фільтр нескінченною імпульсною характеристики виду:

$$H_{pre}(z) = 1 + a_{pre} z^{-1}. \quad (1)$$

Даний фільтр дозволяє «підсилити» високочастотну область спектра сигналу. Це необхідно потрібно для вирівнювання спектра, тому що вокалізовані ділянки мови характеризуються різко спадаючим спектром. І людиною краще сприймаються частоти вище 1кГц. Значення коефіцієнта зазвичай apre вибирається з проміжку [–1.0, –0.4].

На другому етапі мовної сигнал розбивається в часі на короткі проміжки (фрейми), в яких проводиться кепстральний аналіз. Зазвичай тривалість кадру становить від 20 мс до 40 мс. Вважається, що на цих ділянках мовної сигнал можна вважати квазістаціонарним. До фрейму застосовується віконна функція Хеммінга:

$$\omega(n) = 0.54 - 0.46\cos(\frac{2\pi n}{N}). \quad (2)$$

Алгоритм LPCC починається з обчислення p коефіцієнтів $\{a_k\}^p_k$ авторегресійної моделі для кожного фрейма на основі моделі $\hat{S}$:

$$\hat{S}(z) = \frac{A}{1 - \sum_{k=1}^{p} a_k z^{-k}}. \quad (3)$$

Після того, як усі параметри моделі знайдені, обчислюються кепстральних LPCC-коефіцієнти по рекурсивної функції:

$$c(n) = \begin{cases} 0, n < 0 \\ \log_{\varepsilon}(A), n = 0 \\ a_n + \sum_{k=1}^{n-1} (\frac{k}{n}) c(k) a_{n-k}, 0 < n < p \\ \sum_{k=n-p}^{n-1} (\frac{k}{n}) c(k) a_{n-k}, n > p \end{cases} \quad (4)$$

На основі кінцевого числа коефіцієнтів лінійного передбачення може бути отримано нескінченне число LPCC-коефіцієнтів. Встановлено, що 12-20 коефіцієнтів достатньо для формування оптимального для даного методу вектора ознак. Таким чином, отримавши кепстральні коефіцієнти, які необхідно запам'ятати і використовувати для порівняння в процедурі автентифікації.

Також до потужних методів, заснованих на кепстральному аналізі сигналу, відносяться: метод коефіцієнтів перцептивного лінійного передбачення (PLP) і робасних PLP (PLP-RASTA).

4. ВИЯВЛЕННЯ ЛОКАЛЬНИХ ОСОБЛИВОСТЕЙ БІОМЕТРИЧНИХ ОБРАЗІВ НА ОСНОВІ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ

Для проведення автентифікації по біометричному образу, головною задачею є виділення факторів або так названих локальних особливостей зображень, що унікально характеризували би образ користувача.

Одним з найбільш поширених і ефективних методів знаходження таких факторів є метод головних компонент або компонентний аналіз, що знайшов широке застосування у задачах стиснення даних, розпізнавання образів, кодування, обробки зображень, спектрального аналізу і відомий також в теорії розпізнавання образів як перетворення Карунена-Лоева [8, 9].

Однак якщо обробка даних повинна проводитися у реальному часі, на перший план виходять нейромережеві технології, серед яких слід відмітити правило самонавчання та нейрон Е. Оя. На рис. 7 наведено структуру нейрона Оя [10, 11].

За допомогою правила Оя у вигляді [12]:

$$\begin{cases} w_1(k+1) = w_1(k) + \eta(k) y_1(k)(\tilde{x}(k) - w_1(k) y_1(k)), \\ y_1(k) = w_1^T(k)\tilde{x}(k), \; w_1(0) \neq 0 \end{cases} \quad (5)$$

може бути виділено першу головну компоненту, що забезпечує мінімум критерію

$$E_1^k = \frac{1}{k}\sum_{p=1}^{k} (w_1^T \tilde{x}(p))^2. \quad (6).$$

Далі, як і в процедурі стандартного аналізу головних компонент, з кожного вектора $\tilde{x}(k), k = 1,2,\ldots,N$ віднімається його проекція на першу головну компоненту і обчислюється перша головна компонента різниць, що є другою головною компонентою вихідних даних і ортонормальною першій. Третя головна компонента обчислюється шляхом проекції кожного вихідного вектора $\tilde{x}(k)$ на перші дві компоненти, віднімання цієї проекції з $\tilde{x}(k)$ і находження першої головної компоненти різниць, що є третьою головною компонентою первинного масиву даних. Головні компоненти, що залишилися, обчислюються рекурсивно згідно з описаної стратегії.

Таким чином розглянутий метод дозволяє виділити локальні особливості з біометричного образу, занести його в базу даних як еталонний, та використовувати при автентифікації користувача. Такий метод в значній мірі спрощує та підвищує надійність автентифікації користувачів та дозволяє проводити обробку зображення у реальному часі.

ВИСНОВКИ

У статі проведено аналіз проблемних питань систем автентифікації користувачів. Основною метою аналізу є визначення перспективної моделі біометричної автентифікації та ідентифікації особи для впровадження та застосування в Україні. Розглянуто метод виявлення локальних

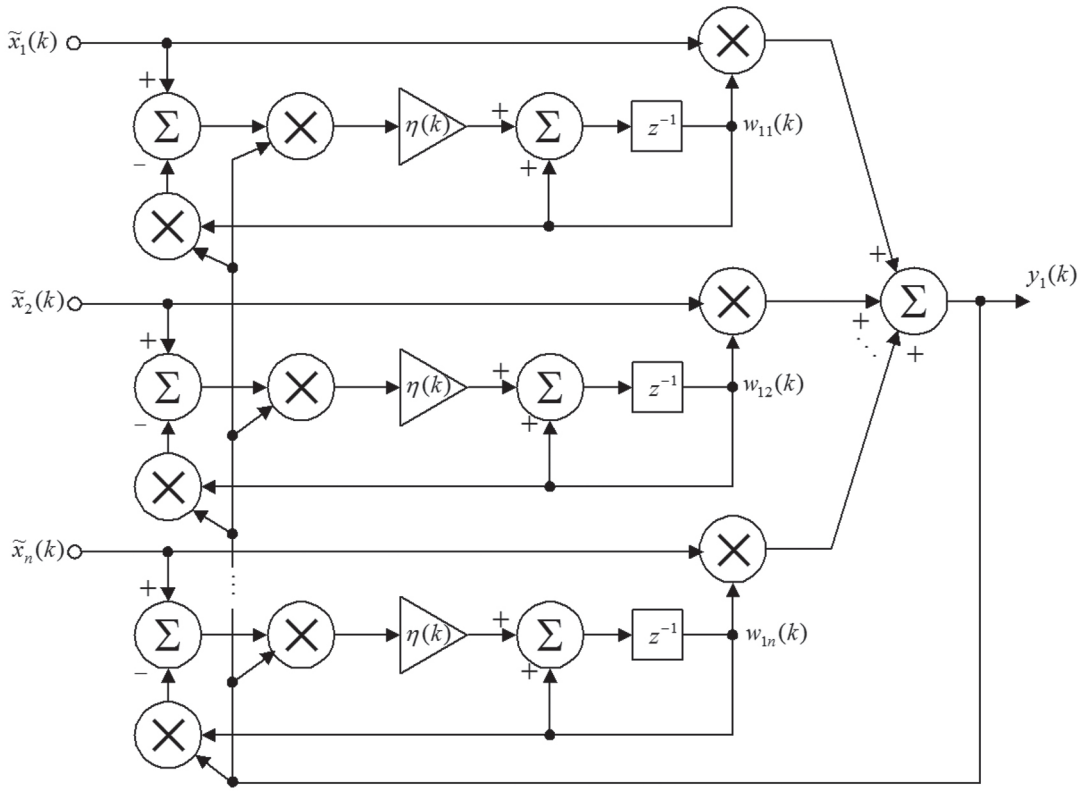


Рис. 7. Нейрон Оя

особливостей біометричних образів за допомогою гібридних штучних нейронних мереж.

Література

[1] *Зубов, Г.Н.* Состояние и перспективы голосовой биометрии [Электронный ресурс] / Г.Н. Зубов, М.В.Хитров // 2007: http://www.chip-news.ru/archive/chipnews/200710/Article_12.pdf

[2] Biometric Standards Activity. [Електронний ресурс] Точка доступу: http://www.biometrics.org/ standards.php.

[3] ANS X9.84:2000. Biometrics Information Management and Security For The Financial Services Industry

[4] Characteristics of biometric systems. Точка доступу: http://www.ccert.edu.cn/education/cissp/ hism/039-041.html

[5] ISO/IEC 19794-2. Finger Minutiae Data.

[6] *Шарий, Т.В.* О проблеме параметризации речевого сигнала в современных системах распознавания речи / Т.В Шарий // Журн. Вісник Донецького національного університету. – 2008. - Сер. А: Природничі науки. - 2. – С. 10-15.

[7] *Doddington, G.R.* Speaker recognition - Identifying people by their voices / G. R. Doddington // Proc. IEEE. – 1985. – 73. – P. 1651-1664.

[8] *Фукунага К.* Введение в статистическую теорию распознавания образов / Фукунага К. – М.: Наука,1979. – 368 с.

[9] *Патрик Э. А.* Основы теории распознавания образов / Патрик Э. А. – М.: Сов. радио, 1980. – 408 с.

[10] *Oja E.* A simplified neuron model as a principal component analyzer / Oja E. // J. of Math. Biology. – 1982. – 15. – P. 267-273.

[11] *Oja E.* Neural networks, principal components, and subspaces / Oja E. // Int. J. of Neural Systems. – 1989. – 1. – P. 61-68.

[12] *Бодянский Е.В.* Искусственные нейронные сети: архитектуры, обучение, применения / Бодянский Е.В., Руденко О.Г. - Харьков: ТЕЛЕТЕХ, 2004. – 369 с.

Поступила в редколлегию 27.04.2012

Філоненко Павло Олександрович, магістрант групи БДІРм-11-1 ХНУРЕ. Область наукових інтересів: дослідження механізмів систем біометричної автентифікації, інформаційні технології.

Барсуков Євген Ігорович, магістрант групи БІКСм-11-1 ХНУРЕ. Область наукових інтересів: дослідження засобів контролю доступу за допомогою біометричних систем по речовому сигналу.

Винокурова Олена Анатоліївна, фото та відомості про автора див. на с. 254.

УДК 343.982.3, 004.89, 004.032.26

Анализ биометрических интеллектуальных методов аутентификации и идентификации личности по отпечаткам пальцев и по голосу для защиты от несанкционированного доступа / П. А. Филоненко, Е. И. Барсуков, Е. А. Винокурова // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 267–274.

В статье приводятся результаты анализа проблемных вопросов систем аутентификации личности. Основной целью анализа является определение перспективной модели биометрической аутентификации и идентификации личности для внедрения и применения в Украине. К основным задачам относятся анализ существующих методов биометрической аутентификации и идентификации личности по отпечаткам пальцев и по голосу для повышения устойчивости от угроз в сфере информационной безопасности. Рассмотрено метод выявления локальных особенностей биометрических образов с помощью гибридных искусственные нейронных сетей.

Ключевые слова: биометрия, аутентификация, гибридные интеллектуальные методы, искусственные нейронные сети.

Рис. 7. Библиогр.: 12 наим.

UDC 343.982.3, 004.89, 004.032.26

Analysis of biometric person authentication and identification intelligent methods by fingerprints and voice for protection from unauthorized access / P.O. Filonenko, E. I. Barsukov, O. A. Vynokurova // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 267–274.

The paper presents the results of analyzing problems of systems of person authentication. The main purpose of the analysis is to identify promising models of biometric person authentication and identification for introducing and using in Ukraine. The main tasks include analysis of existing methods of biometric authentication and identification by fingerprints and a voice to enhance the stability to threats in information security. The local feature detection method for biometric patterns based on hybrid artificial neural networks is considered.

Keyword: biometrics, authentication, hybrid intelligent methods, artificial neural networks.

Fig.: 07. Ref.: 12 items.

МЕТОДЫ И СРЕДСТВА ЗАЩИТЫ ИНФОРМАЦИИ

УДК 638.235.231

ФОРМАЛЬНЫЕ ОСНОВЫ МЕТОДОВ БЛОКИРОВКИ АППАРАТНЫХ ЗАКЛАДНЫХ УСТРОЙСТВ

В.А. ГОРБАЧЕВ

Рассматривается классификация аппаратных закладных устройств, анализ современных методов их обнаружения и предлагается формальная основа методов проектирования сложных электронных систем, которые блокируют аппаратные закладные устройства.

Ключевые слова: модель аппаратной закладки, классификация аппаратных закладок, модель операции доступа, функции управления доступом, оператор сопряжении, оператор управления соединением элементов ЭС.

ВВЕДЕНИЕ

Высокие экономические затраты вынуждают компании по производству электронной продукции использовать интеллектуальную собственность, средства и технологии сторонних разработчиков. Все это приводит к тому, что сложные ЭС разрабатываются и производятся в сравнительно ненадежной рабочей среде [1-3, 5] и могут быть инфицированы аппаратным вирусом (аппаратным закладным устройством). Аутсорсинг в сфере производства ЭС представляет собой серьезную угрозу, особенно для правительственных учреждений, для военной, финансовой, энергетической и политической сферы. ЭС, содержащие аппаратные закладные устройства или просто аппаратные закладки, показали, что они способны выключать центральный процессор, передавать конфиденциальную информацию и обходить программные механизмы аутентификации пользователя. Таким образом, методы для обнаружения аппаратных закладных устройств находятся в центре внимания исследования безопасности IT-систем.

В работе рассматривается подход, использовании которого при проектировании сложных ЭС, позволит ЭС самостоятельно, в реальном масштабе времени, блокировать воздействие аппаратной закладки. Формальная основа, предлагаемого подхода, использует объектно-субъектную модель аппаратной закладки и концепцию управления доступом к ресурсам с учетом критериев гарантированного выполнения политики безопасности.

Оставшаяся часть работы систематизирована, как изложено ниже. В разделе 1 предлагается классификация аппаратных закладок. Анализ существующих методов обнаружения аппаратных закладок приведен в разделе 2. В разделе 3 приводятся формальные основы концепции управления доступом к ресурсам ЭС. Предложенная функция управления доступом может быть использована при разработке методов, которые могут блокировать действие аппаратной закладки в реальном масштабе времени. Выводы по работе представлены в разделе 4.

1. КЛАССИФИКАЦИЯ АППАРАТНЫХ ЗАКЛАДНЫХ УСТРОЙСТВ

Введем определение АЗ. Государственный стандарт [4] определяет аппаратную закладку (АЗ), как скрытно установленное техническое устройство, которое создаёт угрозу безопасности информации.

Классификация аппаратных закладок могут быть выполнена на основании трех главных параметров [6, 7, 10]: физические характеристики, условия активации, функциональные характеристики.

Физические характеристики: Физические характеристики АЗ могут быть либо **функционального**, либо **параметрического** типа**.** Если АЗ изменяет функции системы, в этом случае тип АЗ является функциональным. Тип АЗ будет параметрическим, если ее действие приводит к уменьшению надежности функционирования за счет повышению температуры ЭС, снижению напряжения питания и т.д.

Еще одной физической характеристикой является **размер** АЗ. АЗ может быть большого размера, если она состоит из нескольких компонентов, которые в свою очередь распределены в пределах системы. Если АЗ соизмерима с несколькими транзисторами, в этом случае размер ее маленький.

Характеристики активации: Рассмотрим два типа активации. Внутренний тип активации предусматривает возможность запуска АЗ внутренними датчиками, внутренними состояниями некоторого процесса, определенным шаблоном ввода или внутренним счетчиком. Запуск АЗ извне предполагает использование антенны или других датчиков, доступных злоумышленнику.

Функциональное назначение. Проведем классификацию АЗ по их функциональному назначению [10]: накапливающего типа; разрушающего или блокирующего типа; модифицирующие протокол передачи данных. С точки зрения информационной безопасности, эти АЗ нарушают конфиденциальность, целостность и доступность информации.

2. АНАЛИЗ МЕТОДОВ ОБНАРУЖЕНИЯ АЗ

В этом разделе анализируются современные методы обнаружения АЗ. На рис. 1 приведена классификация методов обнаружения АЗ.

Методы обнаружения АЗ можно классифицировать по двум основным типам: разрушающие и неразрушающие исследуемую ЭС [8, 11].

Разрушающие методы, описанные в работах [12, 13, 14], используют эталон изготовленной ИС, который подлежит деметаллизации с использованием химико-механической полировки, а затем сканирующего электронного микроскопа, получая изображение для реконструкции и анализа. Однако, такой подход является чрезвычайно дорогостоящим и трудоемким, а также плохо переносим для случаев, когда увеличивается плотность интеграции ИС. Кроме того, результаты анализа образцов не могут распространяться на все изготавливаемое количество ЭС.

Неразрушающий метод может быть, в свою очередь, классифицирован на два основных типа: встроенный и невстроенный. Невстроенный метод оставляет оригинальный проект ЭС неизменным, когда при встроенном методе проект ЭС изменяется для внедрения дополнительных компонентов, направленных на обнаружение АЗ.

Встроенные методы обнаружения АЗ. Эти методы могут быть представлены тремя классами: профилактические методы, направлены, на то, чтобы не допустить установку АЗ во время проектирования или изготовления ИС; методы реального времени, устойчивые к воздействию АЗ; вспомогательные методы, облегчающие обнаружение и блокировку установленных АЗ на ЭС.

Встроенные профилактические методы. В работе [14] было отмечено, что внедрение АЗ зависит от наличия свободного «мертвого» пространства на макете ИС. Владея определенной информацией, злоумышленник способен, используя оптимизацию логики и более оптимальный метод размещения, освободить пространство для АЗ. Методики автоматизированного проектирования, предложенные в работе [15], необходимы для предотвращения успешной установки АЗ. Здесь сам оригинальный проект существенно усложняет для злоумышленника возможность расширения пространства для АЗ.

Настоящая работа посвящена новому подходу, который объединяет группу методов (устойчивые к АЗ рис. 1), которые реализует заданную политику безопасности, основанную на стратегии управления доступом к компонентам ЭС. Заданная политика безопасности обеспечивается на архитектурном уровне с помощью специальных методов проектирования ЭС.

Частным случаем предыдущего подхода является группа вспомогательных методов. Методы этой группы используют различные частные решения, зависящие от структуры конкретной ЭС. Например, в работе [16] изменение напряжения питания альтернативных логических уровней в ИС приводит к активности ранее запущенных АЗ. В работе [17] авторы предлагают метод изменения задержки распространения сигналов. В проверенную цепь ЭС вводится задержка, которая позволяет обнаружить присутствие АЗ по времени изменения распространения сигналов.

В невстроенных методах АЗ обнаруживается путем сравнения тестируемой ИС с оригинальной ИС или с ее функциональной моделью. Они могут быть разделены на два основных типа: оперативные и тестовые. В оперативном методе

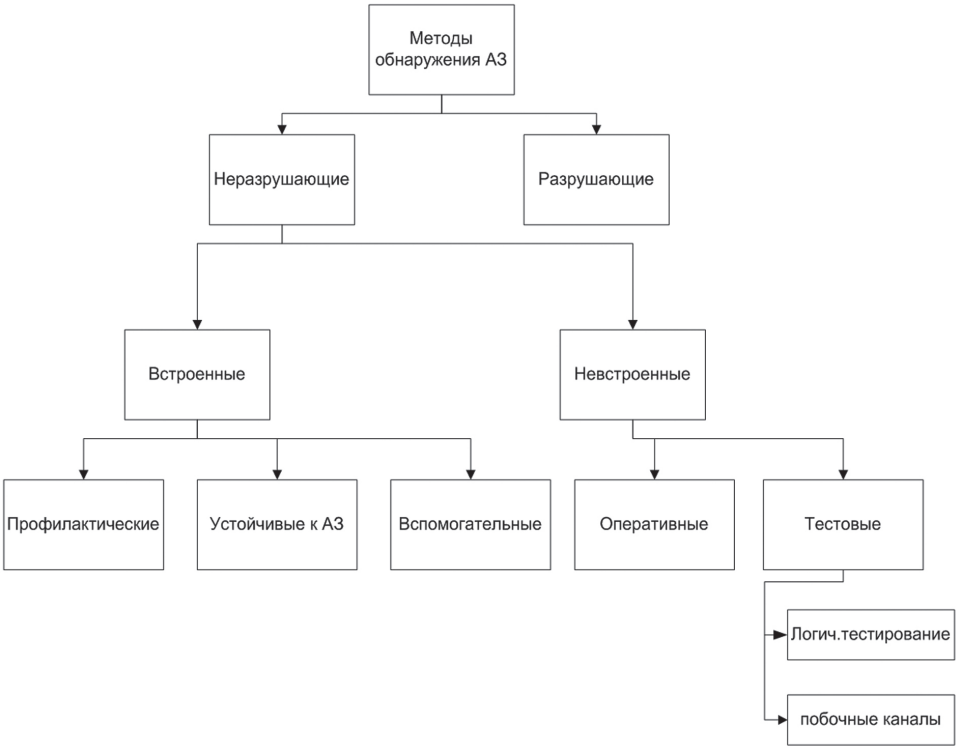


Рис. 1. Классификация методов обнаружения АЗ

используют систему наблюдения, которая пытается выявлять подозрительные действия при работе в реальных условиях, в то время как тестовые методы, направлены на обнаружение инфицированных чипов до их внедрения.

Невстроенные оперативные методы обнаружения АЗ. В работе [18] авторы предлагают включения в систему дополнительного перенастраиваемого устройства, логика которого включает наблюдение за ЭС в режиме реального времени. Проверки могут быть выполнены одновременно с работой схемы в реальных условиях с применением соответствующих мер, когда обнаруживается отклонение от нормального функционирования. Однако, эффективность и аппаратные расходы, связанные с этим методом в работе не упоминаются. В работе [19] авторы предлагают новую SoC архитектуру шины, которая может обнаружить вредоносное поведение шины, связанное с АЗ, защитить систему и системные шины от них и докладывать о вредоносном поведении в системный процессор без потери производительности самой шины.

Невстроенные оперативные методы обычно характеризуются значительной производительностью и энергопотреблением, однако они способны обеспечить 100% уверенность в ожидаемых результатах.

Невстроенные методы обнаружения АЗ в тестовом режиме. Существуют два основных подхода тестирования для обнаружения АЗ: основанные на логике тестирования, и те, которые основаны на измерении параметров побочных каналов, например, таких как мощность, задержка и т.д. Основным недостатком невстроенных методов является требование оригинальной промышленной ИМС или функциональной модели, а также достаточно сложные аппаратно-программные комплексы тестировании.

Подходы, основанные на логике тестирования: В работе [8] описан двухуровневый метод формирования тестовых последовательностей, как объединение стохастических источников по времени и множеству входов. Преимущество метода состоит в том, что он позволяет учитывать специфицированные протоколы обмена информацией объекта исследования с внешней средой при формировании тестовых последовательностей. Это свойство повышает вероятность активации АЗ. Недостатком предложенного метода является неточные оценки полноты тестирования.

Подход, основанный на анализе побочных каналов. Анализ побочных каналов, основан на выявлении АЗ посредством наблюдении их влияния на физические параметры, такие как замыкания цепи, потребляемая мощность или задержка.

В работе [20] авторы ввели понятие дактилоскопии ИС, где каждый экземпляр ИС связан с подписью, называемой «отпечаток пальцев», которая получается путем измерения одного или нескольких параметров побочных каналов. Из анализа «следов» используемых в качестве отпечатков пальцев ИС в этой работе, авторы смогли обнаружить АЗ размером 0,01% от общего размера цепи. Задержки распространения выходных портов были использованы в работе [22] как «отпечатки пальцев», с широкими характеристиками для изменения параметров процесса.

Преимущество этого подхода состоит в том, что если даже во время тестового испытания не удалось обнаружить АЗ, ее наличие может быть обнаружено с помощью некоторых параметров побочных каналов. Тем не менее, основные проблемы анализа побочных каналов связаны с большим разнообразием процессов в современных нано технологиях и помех при измерениях, которые могут маскировать последствие от АЗ, особенно для АЗ маленького размера.

Проведённый анализ угроз, исходящих от АЗ, и методов защиты от них показал, что

1) в настоящий момент не существует единого метода, который мог бы применяться для выявления любых классов АЗ;

2) в работах, посвященных рассматриваемой теме, практически, не рассматриваются формальные модели АЗ и методы борьбы с ними. Это не позволяет рассмотреть проблему с общих позиций.

3. ФОРМАЛЬНЫЕ ОСНОВЫ МЕТОДОВ, КОТОРЫЕ БЛОКИРУЮТ ДЕЙСТВИЕ АЗ В РЕАЛЬНОМ МАСШТАБЕ ВРЕМЕНИ

В теории компьютерной безопасности формальное моделирование политики безопасности является одним из методов, который позволяет оценить эффективность различных аспектов противодействия угрозам и обеспечить эффективные средства защиты формально подтвержденной алгоритмической базой.

Успешная разработка модели безопасности зависит от качества используемой модели самой ЭС, а также от моделей угроз АЗ. Формальные модели АЗ различных классов, использующие понятия объект, субъект, операции доступа для пары «объект-субъект» рассмотрены в [22].

Вышеупомянутым абстрактным понятиям поставим в соответствие такие физические представления в среде ЭС:

Объект (O_j) – часть ресурсов системы, находящаяся в дискретный момент времени в пассивном состоянии относительно информации, а также других аппаратных элементов этой системы.

Субъект (S_i) – электронный компонент, находящийся в дискретный момент времени в активном состоянии, а именно способный осуществить доступ к объекту. Будем рассматривать такой компонент, как пару: ресурс (объект, субъект) и доступ.

Следует отметить, что практически все компоненты ЭС в различные моменты времени могут выполнять функции хранения, получения, обработки и передачи информации. Следовательно,

одно и то же устройство может быть объектом, например, как в первом случае, или субъектом, в остальных случаях.

Рассмотрим понятие пользователя (злоумышленника) в рамках объектно-субъектного подхода, сформулируем важное свойство пользователя (злоумышленника) в виде следующей аксиомы.

Аксиома. Пользователь (злоумышленник) воспринимает объекты и получает информацию о состоянии ЭС через элементы, которые он должен активизировать, т.е. через субъекты.

Таким образом, для реализации своих целей, пользователь (злоумышленник) должен перевести некоторый элемент системы в активное состояние. Очевидно, что для злоумышленника таким компонентом системы будет АЗ.

Важно отметить, что, в отличие от злоумышленника, пользователь – физическое лицо, аутентифицируемое некоторой информацией и управляющее субъектом(ми) ЭС, использует только штатные ресурсы системы.

Пользователь (злоумышленник) является внешним субъектом или субъектом внешней среды ЭС.

Понятие доступа является одним из основополагающих в теории защиты информации, поскольку разрешение или запрет доступа для заданных множеств субъектов и объектов в конечном итоге определяет безопасность ЭС. Формализуем операцию доступа субъекта к объектам, как категорию субъектно-объектной модели.

Сначала, работу механизма доступа продемонстрируем на следующем примере. Допустим, некоторому процессу, протекающему в ЭС, необходимо прочитать данные с накопителя на жёстком диске. Для этого процесс в некоторый момент времени t, обращается к контролеру жесткого диска с соответствующим запросом. В этот момент времени устройство, которое инициирует получение данных, является субъектом, а контроллер жёсткого диска - объектом. При этом субъект изменяет содержимое внутренних регистров объекта, тем самым, изменяя его свойства и порождая новый субъект в ЭС. В следующий момент времени (t+1), порождённый субъект обращается к следующему объекту (непосредственно к контроллеру, управляющему механикой жёсткого диска) и передаёт ему соответствующие запросы, изменяя структуру последнего и тем самым, порождая новый субъект и т.д. Обобщим этот пример на функционирование АЗ.

Очевидно, что при активации и функционировании АЗ, в зависимости от ее типа, в системе будут порождаться неспецифицированные потоки команд (P') и данных (P''). Рассмотрим случаи их возникновения.

Предположим, что в системе от объекта O_j к объекту O_m создается специфицированный поток данных $P''_{\text{сп}}$. Для выполнения этой операции в объекте O_j необходимо активизировать (создать) субъект S_j, который, для данной операции, будет специфицированным субъектом системы. Чтобы активизировать в объекте O_j субъект S_j необходим специфицированный субъект S_i, не принадлежащий объекту O_j, который выполнит операцию с помощью специфицированного потока команд $P'_{\text{сп}}$. Этот процесс описывается парой операций доступа [22]:

$$\begin{aligned} &Create\,(S_i,\, O_j,\, P'_{\text{сп}}) \rightarrow S_j, \\ &Stream\,(S_j,\, O_j,\, P''_{\text{сп}}) \rightarrow O_m. \end{aligned} \qquad (1)$$

Если субъект S_i играет роль нарушителя, а субъект S_j – роль АЗ, тогда процесс описывается следующей парой операций доступа:

$$\begin{aligned} &Create\,(S_i, O_j,\, P') \rightarrow S_{A3}, \\ &Stream\,(S_{A3},\, O_j,\, P'') \rightarrow O_m. \end{aligned} \qquad (2)$$

Очевидно, что для данного процесса имеют место неспецифицированные потоки команд P' и данных P''.

Используя субъектно-объектную модель АЗ, а также то, что, согласно аксиоме [23], все вопросы безопасности информации описываются доступами субъектов к объектам, с целью разработки модели ПБ, введем понятие **функции управления доступом.** Эта функция будет обеспечивать реализацию ПБ на логическом уровне.

Пары (S_i, O_j) связываются множеством разрешенных операций $P'_{\text{сп}}$. Это множество определяется ПБ и является подмножеством всего множества P возможных операций для этой пары. В то же время, пары (S_i, O_j) могут связываться множеством запрещенных, с точки зрения ПБ, операций P'. Задачей ПБ является контроль и блокирование выполнения операций из множества P'. Очевидно, что $P = P'_{\text{сп}} \cup P'$.

Если учесть предположение о том, что АЗ может воспользоваться штатным каналом ЭС, т.е. каналом, который предназначен для поддержки операции доступа из множества P_R', то $P'_{\text{сп}} \cap P' = \varnothing$. Таким образом, становится очевидно, что разрешение либо запрещение самого факта доступа для обеспечения заданной политики безопасности недостаточно.

Анализируя сказанное и операции доступа (2) приходим к важному выводу: для обеспечения гарантированного выполнения заданной политики безопасности **(**ПБ) в ЭС нужно контролировать не только факт доступа субъекта к объекту, но и неспецифицированные потоки команд P' и данных P''.

Для управления операциями доступа, а так же для обнаружения и блокировки неспецифицированных потоков в системе предлагается использовать функцию управления доступом вида:

$$F = F\,(\overline{S},\, \overline{O},\, \overline{P'},\, t)\,. \qquad (3)$$

Аргументами этой функции являются: субъекты, объекты, легализованные операции и время t осуществления доступа.

Определим область определения аргументов функции F. Конкретные значения этих аргументов определяются технической документацией на изделие.

Определим область определения функции управления доступом. Функция управления доступом может быть задана в любом виде, например, в виде таблицы, либо в виде алгоритма. Она может принимать значения 1, если доступ разрешен, и 0, если доступ запрещен.

Для реализации ПБ при управлении доступом на физическом уровне введем понятие оператора управления соединениями элементов ЭС. Теоретико-множественную модель топологии (архитектуры) ЭС представим следующим образом.

Обозначим множество элементов ЭС через $\overline{C} = (C_0, C_1, C_2, ..., C_N)$. Очевидно, что в некоторый момент времени t $\overline{C} = (\overline{O}, \overline{S})$, где $\overline{O}$ — это множество объектов, а множество субъектов в этой системе — $\overline{S}$. Как было отмечено выше, каждый субъект и объект являются ресурсами системы, поэтому $\overline{S} \subseteq \overline{O}$.

Построим теоретико-множественную модель сопряжения элементов сетью каналов связи, обеспечивающих передачу сигналов между элементами.

Вход элемента C_j состоит из m_j входных контактов; контакт $X_i^{(j)}$ принимает элементарные сигналы $x_i^{(j)}(t)$; $i = 1, 2, ..., m_j$; $j = 1, 2, ..., N$. Аналогично выход элемента C_j состоит из r_j выходных контактов; контакт $Y_l^{(j)}$ выдает элементарные сигналы $y_l^{(j)}(t)$; $l = 1, 2, ..., r_j$.

Внешнюю среду можно представить в виде фиктивного элемента C_0, выход которого содержит — r_0 выходных контактов $Y_l^{(0)}$, а его вход состоит из m_0 входных контактов $X_i^{(0)}$.

Изложенные соображения приводят к заключению, что каждый C_j (в том числе и C_0) как элемент ЭС достаточно характеризовать множеством входных портов (контактов):

$$\left\{X_i^{(j)}\right\}=\left(X_1^{(j)}, X_2^{(j)}, \ldots, X_{m_j}^{(j)}\right), j = \overline{0, N},$$

и множеством выходных портов (контактов):

$$\left\{Y_l^{(j)}\right\}=\left(Y_1^{(j)}, Y_2^{(j)}, \ldots, Y_{r_j}^{(j)}\right), j = \overline{0, N}.$$

Другими словами, математической моделью интерфейса элемента C_j, используемой для формального описания его сопряжения с другими элементами системы и внешней средой, является пара множеств: $\left\{X_i^{(j)}\right\}$ и $\left\{Y_l^{(j)}\right\}$.

Рассмотрим множество всех входных контактов всех элементов данной системы и внешней среды $\bigcup_{j=0}^{N}\left\{X_i^{(j)}\right\}, i = \overline{1, m_j}$, а также всех выходных контактов $\bigcup_{j=0}^{N}\left\{Y_l^{(j)}\right\}, l = \overline{1, r_j}$. В силу предположения, что каждому входному контакту $X_i^{(j)}$ соответствует не более чем один выходной контакт $Y_1^{(k)}$, с которым он связан элементарным каналом, можно ввести однозначный оператор сопряжения (отношения) R:

$$Y_1^{(k)} = R(X_i^{(j)}). \qquad (4)$$

с областью определения на множестве $\bigcup_{j=0}^{N}\left\{X_i^{(j)}\right\}, i = \overline{1, m_j}$, и областью значений на множестве $\bigcup_{j=0}^{N}\left\{Y_l^{(j)}\right\}, l = \overline{1, r_j}$. Фактически оператор R **однозначно** сопоставляет входному контакту $X_i^{(j)}$ выходной контакт $Y_1^{(k)}$, которые связываются между собой элементарным каналом. Если в рассматриваемой системе к данному контакту $X_i^{(j)}$ не подключен никакой элементарный канал, то оператор (4) не определен на этом $X_i^{(j)}$.

Совокупность множеств $\left\{X_i^{(j)}\right\}$, $\left\{Y_l^{(j)}\right\}$ и оператора R будем называть схемой сопряжения элементов в системе или **топологической моделью** системы. Рассмотренная формальная модель (4) содержит исчерпывающую информацию о соединениях компонентов системы.

Оператор сопряжения (4) можно задать в виде таблицы. В ней на пересечении строк с номерами элементов системы j и столбцов с номерами выходных контактов i располагаются пары чисел (k, l), указывающие номер элемента k и номер его выходного контакта l, с которым соединен контакт $X_i^{(j)}$.

Как уже было показано выше, модель ЭС, связанная с реализацией ПБ, не укладывается в рамки простой модели взаимодействия электронных компонентов ЭС (4).

Во-первых, в процессе функционирования ЭС структура связей изменятся во времени под управлением выполняемых команд.

Во-вторых, структура связей должна изменяться в соответствии с правилами доступа, т.е. в соответствии с моделью безопасности.

Для учета эти факторов, в оператор сопряжения R введем параметр времени и функцию управления доступом (3):

$$Y_l^{(k)} = K\left(X_i^{(j)}, t, F_i^j\right), \qquad (5)$$

где t — время, F_i^j — функция управления доступом между O_i и S_j.

Фактически, оператор K управляет доступом на физическом уровне, назовем его оператором управления соединением элементов ЭС. Он может принимать значения 1, если для пары элементов (j, k) имеется связь, либо 0, если для этой пары элементов связь отсутствует.

ВЫВОДЫ

Анализируя результаты, полученные в работе, можно сделать следующие выводы.

1. Формальные модели АЗ [24], а также предложенные в настоящей работе функция управления доступом (3) и оператор управления соединением (5), могут быть использованы при построении формальной модели безопасности системы.

2. Формальная модель безопасности системы, основанная на концепции управления доступом, в свою очередь, может быть положена в основу, как архитектуры ЭС, способной блокировать действия АЗ, так и методов ее проектирования.

Литература

[1] DARPA, Arlington, VA, "Trust for integrated circuits," 2007. [Online]. Available: http://www.darpa.mil/MTO/solicitations/baa07-24/index.html

[2] *S.T. King, J. Tucek, A. Cozzie, C. Grier, W. Jiang, and Y. Zhou,* "Designing and implementing malicious hardware," in *Proc. 1st Usenix Workshop Large-Scale Exploits Emergent Threats (LEET)*, San Francisco,CA, 2008, pp. 1–8.

[3] *F. Wolff, C. Papachristou, S. Bhunia, and R. Chakraborty,* "Towards trojan-free trusted ICs: Problem analysis and detection scheme," in *Proc. Des. Autom. Test Euro. (DATE)*, Munich, Germany, 2008, pp. 1362–1365.

[4] ДСТУ 3396.2-97 Захист інформації. Технічний захист інформації. Терміни та визначення. Затверджено наказом Держстандарту України від 11.04.97р. №200.

[5] *Горбачев В.А., Степаненко В.В.* Сертификация периферийных устройств компьютерных систем.// Радиотехника: сб. научн. трудов. Выпуск 134.- Харьков: ХТУРЭ, 2003. С. 206-209.

[6] *Xiaoxiao Wang, Mohammad Tehranipoor and Jim Plusquellic:* Detecting Malicious Inclusions in Secure Hardware, Challenges and Solutions, 1st IEEE International Workshop on Hardware-Oriented Security and Trust (HOST'08), 2008.

[7] *Rajat Subhra Chakraborty, Seetharam Narasimhan and Swarup Bhunia*: Hardware Trojan: Threats and Emerging Solutions, Dept. of Electrical Engineering and Computer Science Case Western Reserve University Cleveland, Ohio, USA, 2010.

[8] *Горбачев В.А., Саранча С.Н., Степаненко В.В.* Сертификация сложных электронных систем с использованием функциональной модели объекта на полном наборе входных слов. Науково-технічний збірник. Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. – Київ: КНУ-КПІ, 2002. Вип. 5. – С.139–144.

[9] *Benjamin Sanno:* Detecting Hardware Trojans, Ruhr-University Bochum, Germany, July 22, 2009.

[10] *Горбачев В.А., Иванисенко И.Н.* Классификация и формальные модели аппаратных закладных устройств. Прикладна радіоелектроніка та інформатика, Харьков: ХТУРЭ. Том 6, 2007. № 2. – С. 306-310.

[11] Hardware Trojan: Threats and Emerging Solutions (Invited Paper) Rajat Subhra Chakraborty, Seetharam Narasimhan and Swarup Bhunia Dept. of Electrical Engineering and Computer Science Case Western Reserve University Cleveland, Ohio, USA, 2009

[12] Chipworks, Inc., "Semiconductor Manufacturing - Reverse Engineering of Semiconductor components, parts and process". [Online]. Available: http://www.chipworks.com

[13] *J.A. Kash, J.C. Tsang and D.R. Knebel,* "Method and Apparatus for Reverse Engineering Integrated Circuits by Monitoring Optical Emission", United States Patent Number 6,496,022 B1, 2002.

[14] *M. Banga and M.S. Hsiao,* "VITAMIN: Voltage Inversion Technique to Ascertain Malicious Insertions in ICs", HOST, 2009.

[15] *R.S. Chakraborty and S. Bhunia,* "Security against Hardware Trojan through a Novel Application of Design Obfuscation", ICCAD, 2009.

[16] *M. Banga and M.S. Hsiao,* "VITAMIN: Voltage Inversion Technique to Ascertain Malicious Insertions in ICs", HOST, 2009.

[17] *J. Li and J. Lach,* "At-Speed Delay Characterization for IC Authentication and Trojan Horse Detection", HOST, 2008.

[18] *M. Abramovici and P. Bradley,* "Integrated Circuit Security - New Threats and Solutions", CSIIR Workshop, 2009.

[19] *L.W. Kim, J.D. Villasenor and C.K. Koc,* "A Trojan-resistant Systemon-chip Bus Architecture", Intl. Conf. on Military Communication, 2009

[20] *D. Agrawal et al,* "Trojan detection using IC fingerprinting", IEEE Symp. on Security and Privacy, 2007.

[21] *Y. Jin and Y. Makris,* "Hardware Trojan Detection Using Path Delay Fingerprint", HOST, 2008.

[22] *Горбачев В.А. Иванисенко И.Н.* Классификация и формальные модели аппаратных закладных устройств. Прикладна радіоелектроніка та інформатика, Харьков: ХТУРЭ. – том 6, 2007. № 2 С. 306-310.

[23] *Щербаков А.Ю.* Введение в теорию и практику компьютерной безопасности. – М.: издатель Молгачёва С.В., 2001. – 352 с., ил.

[26] Canadian Tusted Computer Product Evaluation Criteria, v. 3.0. Canadian System Security Centre, Communications Security Establishment, Government of Canada, 1993.

Поступила в редколлегию 5.03.2012

Горбачев Валерий Александрович, профессор кафедры ЭВМ ХНУРЭ. Область научных интересов: системный анализ.

УДК 638.235.231

Формальні основі методів блокування апаратних закладних пристроїв / В.О.Горбачов // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 275–280.

Розглядається класифікація апаратних закладних засобів, аналіз сучасних методів їх виявлення, та пропонуються формальні основи методів проектування складних електронних систем, що блокують апаратні закладні пристрої.

Ключові слова: модель апаратної закладки, класифікація апаратних закладок, модель операції доступу, функція управління доступом, оператор управління з'єднанням.

Іл. 1. Бібліогр.: 26 найм.

UDC 638.235.231

Formal basis of malicious hardware blocking methods / V.A. Gorbachov / / Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 275–280.

The paper considers the malicious hardware classification, analysis of modern detection methods and proposes a formal basis of methods of designing complex electronic systems that block the malicious hardware.

Keywords: malicious hardware model, malicious hardware classification, access operation model, access control functions, conjugator, electronic system element connection control operator.

Fig. 1. Ref.: 26 items.

УДК 681.142:681.3

МЕТОД ОПЕРАТИВНОГО КОНТРОЛЯ ДАННЫХ В КЛАССЕ ВЫЧЕТОВ НА ОСНОВЕ ИСПОЛЬЗОВАНИЯ ПОЗИЦИОННОГО ПРИЗНАКА НЕПОЗИЦИОННОГО КОДА

С.А. МОРОЗ, В.А. КРАСНОБАЕВ, А.А. ЗАМУЛА

В данной статье рассматривается метод оперативного контроля данных, представленных кодом класса вычетов. Приведены примеры конкретного выполнения операций контроля.

Ключевые слова: немодульные (позиционные) операции, кодовая структура, контроль данных, непозиционная система счисления.

ВВЕДЕНИЕ

Основное преимущество непозиционной системы счисления в классе вычетов (КВ) заключается в возможности организации процесса быстрой реализации следующих модульных операций: арифметические операции сложения, вычитания и умножения; операции логического сложения, вычитания и умножения по модулю два; деление целых чисел и пр. [1, 2]. Однако, в системе передачи и обработки данных (СПОД) общего назначения кроме вышеперечисленных арифметических операций необходимо осуществлять так называемые в КВ немодульные (позиционные) операции. К таким операциям в первую очередь относятся следующие:

– арифметическое и алгебраическое сравнение операндов и их абсолютных величин;

– определение знака операнда;

– определение наличия переполнения разрядной сетки СПОД;

– округление величины результата операции;

– вычисление абсолютной величины числа;

– деление и умножение дробей;

– перевод данных из кода в КВ в позиционную систему счисления (ПСС) и наоборот;

– расширение исходного КВ (это информационный процесс, когда по известным остаткам $\{a_i\}$, соответствующих основаниям $\{m_i\}$, определяются значения остатков этой же кодовой структуры по другим дополнительным основаниям);

– контроль, диагностика и коррекция ошибок и др.

В общем случае все позиционные операции сводятся к процедуре определения номера j числового $[j \cdot m_i, (j+1) \cdot m_i)$ интервала попадания (нахождения) числа $A = (a_1, a_2, ..., a_i, ..., a_n, a_{n+1})$. Для определения номера j числового интервала нахождения числа A целесообразно использовать так называемыми позиционными признаками непозиционного кода (ППНК). Из существующих ППНК чаще всего в КВ используют следующие признаки [1, 3]:

– признаки, основанные на процедуре перевода числа из КВ в ПСС;

– признаки, основанные на процедуре нулевизации (определение значения γ_{n+1});

– признаки, основанные на процедуре расширения системы оснований данного КВ;

– ранг r числа A.

Основными недостатками вышеперечисленных ППНК является, во-первых, техническая и временная сложность их формирования (разработка) для заданной кодовой структуры $A = (a_1, a_2, ..., a_n, a_{n+1})$ данных и, во-вторых, значительное время реализации, посредством существующих позиционных признаков, немодульных операций в КВ, в частности, операции контроля данных [3].

Таким образом, важны исследования, посвященные разработке ППНК в КВ, с помощью которых оперативно реализуются немодульные операции. Отметим, что любая немодульная операция может быть реализована посредством совокупности (последовательности) определенных модульных и немодульных операций, реализующиеся посредством ППНК.

Цель статьи – сформировать ППНК, на основании, которого разработать метод оперативного контроля данных в КВ.

Основная часть. Рассмотрим основные требования к ППНК, на основе которого в дальнейшем будет разработан метод повышения оперативности контроля данных [4, 5]:

– посредством используемого (выбранного, разработанного, сформированного) ППНК необходимо достоверно определить правильность или неправильность числа А в КВ (определить факт нахождения или нет числа А в информационном числовом $[0, M]$ интервале, где $M = \prod_{i=1}^{n} m_i$);

– простота формирования ППНК для заданий кодовой $A = (a_1, a_2, ..., a_n, a_{n+1})$ структуры данных;

– простота использования сформированного признака для проведения контроля данных в КВ;

– признак должен иметь четкий и понятный физический смысл;

– аналитически признак должен описываться не сложным математическим соотношением;

– посредством использования ППНК возможно технически просто реализовать систему контроля (СК) данных в КВ;

– применение выбранного признака непозиционного кода должно обеспечить повышение контроля данных в КВ;

– использование ППНК должно по возможности исключать наиболее сложные позиционные операции из процедуры контроля, диагностики и коррекции ошибок в КВ.

В связи с вышеизложенным целесообразно разработать и исследовать метод оперативного контроля данных в КВ на основе использования ППНК. Вначале рассмотрим процедуру (алгоритм) формирования ППНК, на основе непозиционной кодовой структуры $A = (a_1, a_2, ..., a_i, ..., a_n, a_{n+1})$ данных, представленной в КВ основаниями $\{m_i\}$, $i = \overline{1, n+1}$, так называемого однорядового кода (ОК). В общем виде ОК $K_N^{(n_A)} = \left\{ Z_{N-1}^{(A)} Z_{N-1}^{(A)} ... Z_1^{(A)} Z_0^{(A)} \right\}$ представляет собой последовательность двоичных $Z_K^{(A)}$ $(K = \overline{0, N-1})$ разрядов, состоящую из единиц и только одного нуля, находящегося на n_A-м месте (считая справа, от разряда $Z_0^{(A)}$, налево, до разряда $Z_{N-1}^{(A)}$).

Физически ППНК n_A определяет номер j числового $[j \cdot m_i, (j+1) \cdot m_i)$ числового интервала нахождения числа A. Математически ППНК представляет собой натуральное n_A число, которое указывает на местоположение нулевого двоичного разряда в записи ОК $K_N^{(n_A)} \left(Z_{n_A}^{(A)} = 0 \right)$.

Процедура формирования ОК $K_N^{(n_A)}$ состоит в следующем. Для выбранного основания m_i КВ по значению остатка a_i числа $A = (a_1, ..., a_{i-1}, a_i, a_{i+1}, ..., a_n, a_{n+1})$ в блоке констант нулевизации (БКН) определяется константа вида $KH_{m_i}^{(A)} = \left(a_1^{'}, ..., a_{i-1}^{'}, a_i, a_{i+1}^{'}, ..., a_{n+1}^{'} \right)$.

Далее, посредством выбранной константы нулевизации $KH_{m_i}^{(A)}$, число A смещаем на левый край интервала $[j \cdot m_i, (j+1) \cdot m_i)$ путем реализации операции

$$A_{m_i} = A - KH_{m_i}^{(A)} = \left(a_1, a_2, ..., a_{i-1}, a_i, a_{i+1}, ..., a_{n+1} \right) - \\ - \left(a_1^{'}, a_2^{'}, ..., a_{i-1}^{'}, a_i, a_{i+1}^{'}, ..., a_{n+1}^{'} \right) = \\ = \left[a_1^{(1)}, a_2^{(1)}, ..., a_{i-1}^{(1)}, 0, a_{i+1}^{(1)}, ..., a_{n+1}^{(1)} \right].$$

Очевидно, что число A_{m_i} кратно значению модуля m_i КВ.

Известно, что правильность числа A в КВ определяется попаданием или непопаданием его в числовой информационный $[0, M)$ интервал. Если число A находится вне этого интервала $(A \geq M)$, то оно считается искаженным (неправильным). В этом случае ППНК n_A должен определить факт попадания или непопадания исходного числа A в интервал $[0, M)$. Чтобы определить факт нахождения числа в информационном $[0, M)$ числовом интервале необходимо провести операцию вида $A_{m_i} - K_A \cdot m_i = Z_{K_A}^{(A)}$ (1). Данная операция (1) проводиться одновременно и параллельно во времени посредством совокупности из N констант $K_A \cdot m_i \left(K_A = \overline{0, N-1} \right)$, где $N = \prod_{\substack{K=1 \\ K \neq i}}^{n+1} m_K$:

$$\begin{cases} A_{m_i} - 0 \cdot m_i = Z_0^{(A)}, \\ A_{m_i} - 1 \cdot m_i = Z_1^{(A)}, \\ A_{m_i} - 2 \cdot m_i = Z_2^{(A)}, \\ \quad \dots \\ A_{m_i} - (N-2) \cdot m_i = Z_{N-2}^{(A)}, \\ A_{m_i} - (N-1) \cdot m_i = Z_{N-1}^{(A)}. \end{cases} \quad (2)$$

В этом случае ОК представится в виде $K_N^{(n_A)} = \left\{ Z_{N-1}^{(A)} Z_{N-2}^{(A)} ... Z_1^{(A)} Z_0^{(A)} \right\}$ (3).

В совокупности (2) аналитических соотношений существует единственное значение n_A из (1) для которого $Z_{K_A}^{(A)} = Z_{n_A}^{(A)} = 0$ $(K_A = n_A)$, т.е. $A_{m_i} - n_A \cdot m_i = 0$. Остальные значения (2) равных $Z_l^{(A)} = 1$ $(A_{m_i} - l \cdot m_i \neq 0; \ l \neq n_A)$. В общем случае количество N двоичных разрядов в записи ОК $K_N^{(n_A)}$ равно значению $N = \prod_{\substack{K=1 \\ K \neq i}}^{n+1} m_K$. Отметим, что для определения только факта искажения числа $(A \geq M)$ нет необходимости иметь всю последовательность значений $Z_{K_A}^{(A)}$ совокупности (3). Достаточно иметь ОК $K_{N_i}^{(n_A)}$ длиною всего $N_i =]M / m_i[$ двоичных разрядов (где значение $]M / m_i[$ обозначает целую часть числа M / m_i, его не меньшую). Так как в этом случае значение величин числовых интервалов $[j \cdot m_i, (j+1) \cdot m_i)$, расположенных вне информационного интервала $[0, M)$, не имеют никакого значения для установления факта контроля правильности числа A. Алгоритм формирования ППНК n_A в КВ представлен на рис. 1.

Таким образом, суть метода контроля данных в КВ состоит в следующем (рис. 2). Для контролируемой кодовой структуры $A = (a_1, a_2, ..., a_i, a_n, a_{n+1})$, представленной в КВ, разрабатывается (определяется) ППНК n_A путем формирования ОК $K_{N_i}^{(n_A)} = \left\{ Z_{N_i-1}^{(A)} Z_{N_i-2}^{(A)} ... Z_1^{(A)} Z_0^{(A)} \right\}$ в виде последовательности из N_i двоичных разрядов. Выбор основания m_i КВ производиться специальным образом, в соответствии с определенными критериями. Исходя из значения остатка a_i числа A, выбирается константа нулевизации вида $KH_{m_i}^{(A)} = \left(a_1^{'}, a_2^{'}, ..., a_i, ..., a_n^{'}, a_{n+1}^{'} \right)$. Далее проводиться реализация операции

$$A_{m_i} = A - KH_{m_i}^{(A)} = \left(a_1, a_2, ..., a_i, ..., a_n, a_{n+1} \right) - \\ - \left(a_1^{'}, a_2^{'}, ..., a_i, ..., a_n^{'}, a_{n+1}^{'} \right) = \left[a_1^{(1)}, a_2^{(1)}, ..., 0, ..., a_n^{(1)}, a_{n+1}^{(1)} \right].$$

Используя N_i констант $K_A \cdot m_i$ $\left(K_A = \overline{0, N_i - 1} \right)$ одновременно проводятся операции вычитания $A_{m_i} - K_A \cdot m_i$, в результате которых образуется значение двоичных разрядов $Z_{K_A}^{(A)}$, т.е. формируется ОК $K_{N_i}^{(n_A)}$. Значение ППНК n_A определяется

из равенства $A_{m_i} - n_A \cdot m_i = 0$. Если $n_A > N_i$, то считается что число A – неправильное число. В противоположном случае ($n_A \le N_i$) число A – правильное.

Рассмотрим примеры реализации метода контроля для конкретного КВ, который задан основаниями $m_1 = 3$, $m_2 = 4$, $m_3 = 5$, $m_4 = 7$ и $m_k = m_{n+1} = m_5 = 11$. Данный КВ обеспечивает обработку данных в однобайтовой (l = 1) разрядной сетке СПОД. При этом $M = \prod_{i=1}^{4} m_i = 420$, $M_0 = M \cdot m_{n+1} = 4620$; $N_i = N_{n+1} =]M/m_i[=]M/m_{n+1}[=]420/11[=]38,18[= 39$.

В табл. 1 приведено содержимое блока констант нулевизации относительно основания $m_K = m_{n+1} = 11$.

Таблица 1

Константы $KH_{m_{n+1}}^{(A)}$ нулевизации

Остаток $a_к = a_{n+1}$	Константы нулевизации				
	$m_1 = 3$	$m_2 = 4$	$m_3 = 5$	$m_4 = 7$	m_k=m_5=11
	a'_1	a'_2	a'_3	a'_4	a_5
0000	00	00	000	000	0000
0001	01	01	001	001	0001
0010	10	10	010	010	0010
0011	00	11	011	011	0011
0100	01	00	100	100	0100
0101	10	01	000	101	0101
0110	00	10	001	110	0110
0111	01	11	010	000	0111
1000	10	00	011	001	1000
1001	00	01	100	010	1001
1010	01	10	000	011	1010

1 Выбор информационных $\{m_i\}$, $i = \overline{1,n}$ и контрольного $m_K = m_{n+1}$ ($m_i < m_{i+1}$) оснований КВ для представления данных

$$A = (a_1, a_2, \ldots, a_{i-1}, a_i, a_{i+1}, \ldots, a_n, a_{n+1}),$$

$$\text{НОД}(m_i, m_j) = 1,\ i \neq j.$$

2 Выбор основания $m_i \in \{m_j\}$, $(j = \overline{1, n+1})$ по которому определяется номер j числового интервала $[j \cdot m_i, (j+1) \cdot m_i)$ нахождения числа A.

3 По значению остатка a_i (по модулю m_i) числа A определение константы нулевизации

$$KH_{m_i}^{(A)} = (a'_1, a'_2, \ldots, a'_{i-1}, a_i, a'_{i+1}, \ldots, a'_n, a'_{n+1}).$$

4 Определение значения $A_{m_i} = A - KH_{m_i}^{(A)} =$

$$= (a_1, a_2, \ldots, a_{i-1}, a_i, a_{i+1}, \ldots, a_n, a_{n+1}) -$$

$$-(a'_1, a'_2, \ldots, a'_{i-1}, a_i, a'_{i+1}, \ldots, a_n, a_{n+1}) =$$

$$= [a_1^{(1)}, a_2^{(1)}, \ldots, a_{i-1}^{(1)}, 0, a_{i+1}^{(1)}, \ldots, a_n^{(1)}, a_{n+1}^{(1)}].$$

5 Определение ОК

$$K_N^{(n_A)} = \{Z_{N-1}^{(A)}\ Z_{N-2}^{(A)} \ldots Z_0^{(A)}\},$$

$$K_{N_i}^{(n_A)} = \{Z_{N_i-1}^{(A)}\ Z_{N_i-2}^{(A)} \ldots Z_0^{(A)}\}.$$

$$N = \prod_{\substack{K=1 \\ K \neq i}}^{n+1} m_K, \qquad N_i =]M / m_i[, \qquad M = \prod_{i=1}^{n} m_i.$$

$$A_{m_i} - K_A \cdot m_i = Z_{K_A}^{(A)}.$$

5

$$\begin{cases} A_{m_i} - 0 \cdot m_i = Z_0^{(A)}, \\ A_{m_i} - 1 \cdot m_i = Z_1^{(A)}, \\ \ldots \\ A_{m_i} - (N-2) \cdot m_i = Z_{N-2}^{(A)}, \\ A_{m_i} - (N-1) \cdot m_i = Z_{N-1}^{(A)}. \end{cases}$$

$$\begin{cases} A_{m_i} - 0 \cdot m_i = Z_0^{(A)}, \\ A_{m_i} - 1 \cdot m_i = Z_1^{(A)}, \\ \ldots \\ A_{m_i} - (N_i-2) \cdot m_i = Z_{N-2}^{(A)}, \\ A_{m_i} - (N_i-1) \cdot m_i = Z_{N_i-1}^{(A)}. \end{cases}$$

6 Определение ППНК n_A числа A, т.е. числового значения n_A для которого $Z_{K_A}^{(A)} = Z_{n_A}^{(A)} = 0$,

т.е. $A_{m_i} - n_A \cdot m_i = 0$.

При этом $Z_i^{(A)} = 1, (A_{m_i} - l \cdot m_i \neq 0; l \neq n_A)$.

Рис. 1. Алгоритм формирования ППНК в КВ

Алгоритм определения ОК

$$K_{N_i}^{(n_A)} = \{Z_{N_i-1}^{(A)}\ Z_{N_i-2}^{(A)} \ldots Z_1^{(A)}\ Z_0^{(A)}\}$$

числа $A = (a_1, a_2, \ldots, a_{i-1}, a_i, a_{i+1}, \ldots, a_n, a_{n+1})$.

Алгоритм определения ППНК n_A: $A_{m_i} - n_A \cdot m_i = 0$,

$$Z_{n_A}^{(A)} = 0;\ Z_l^{(A)} = 1,\ A_{m_i} - l \cdot m_i = 1;\ l \neq n_A.$$

Контроль данных

$A = (a_1, a_2, \ldots, a_{i-1}, a_i, a_{i+1}, \ldots, a_n, a_{n+1})$ в КВ. Если $n_A > N_i$, то число Ã неправильное (искаженное). Если $n_A \le N_i$, то число A правильное (неискаженное).

Рис. 2. Метод контроля данных в КВ

Пример 1. Провести контроль данных, представленных в виде A=(01,00,000,010,0001). По значению остатка $a_K = a_{n+1} = a_5$=0001 числа A в БКН (табл. 1) выбирается константа $KH^{(A)}_{m_{n+1}} = (01,01,001,001,0001)$ нулевизации. Далее определяем $A_{m_{n+1}} = A - KH^{(A)}_{m_{n+1}} = (00,11,100,001,0000)$.

Посредством реализации соотношения (2) формируем ОК $K^{(n_A)}_{N_i} = K^{(9)}_{39} = \{11...110111111111\}$. Исходя из вида ОК и используя выражение $A_{m_{n+1}} - n_A \cdot m_{n+1} = 0$, определяем, что

$$n_A = 9 \ (A_{m_{n+1}} - n_A \cdot m_{n+1} = 99 - 9 \cdot 11 = 0),$$

т.е. $Z^{(A)}_{n_A} = Z^{(A)}_9$. Так, как $N_i = 39 > n_A = 9$, то ошибки в данных нет.

Проверка: $A = 100 < M = 420$ (число A правильное).

Пример 2. Провести контроль данных A = (00, 01, 000, 010, 1010). По значению a_5 = 1010 в БКН (табл. 1) выбирается константа вида $KH^{(A)}_{m_{n+1}} = (01,10,000,011,1010)$. Получим, что

$$A_{m_{n+1}} = A - KH^{(A)}_{m_{n+1}} = (10,00,000,110,0000).$$

Так как $A_{m_{n+1}} - n_A \cdot m_{n+1} = 440 - 44 \cdot 11 = 0$, то ОК имеет вид

$$K^{(n_A)}_{N_i} = K^{(40)}_{39} = \{11...11...11\} \text{ и } n_A = 40.$$

Так как $N_i = 39 < n_A = 40$, то ошибка в данных присутствует.

Проверка: $A = 450 > M = 420$ (число A неправильное).

Пример 3. Провести контроль данных A = (01, 11, 010, 000, 1001). По значению a_5 = 1001 в БКН (табл. 1) выбирается константа $KH^{(A)}_{m_{n+1}} = (00,01,100,010,1001)$. Определим что

$$A_{m_{n+1}} = A - KH^{(A)}_{m_{n+1}} = (01,10,011,101,0000).$$

Так как $A_{m_{n+1}} - n_A \cdot m_{n+1} = 418 - 38 \cdot 11 = 0$, то ОК имеет вид

$$K^{(n_A)}_{N_i} = K^{(38)}_{39} = \{011...11...11\} \text{ и } n_A = 38.$$

Исходя из того, что $n_A = 38 < N_i = 39$ делается вывод: число А правильное (не искажено). Однако проверка показывает, что $A = 427 > M = 420$, т.е. A неправильное число (рис. 3).

Ошибка контроля данных (низкая достоверность контроля) в примере 3 вызвана наличием остатка $\alpha = M_i / m_{n+1}$. В св-ою очередь наличие остатка α определяется фактом не кратности значения M_i контрольному модулю m_{n+1}. Для рассмотренного числового [418, 429) диапазона совокупность неправильных $\tilde{A}$ чисел воспринимается как набор правильных A чисел (табл. 2).

Таблица 2

Совокупность кодовых слов

Числовой диапазон [418, 429]	
Правильные числа А	Совокупность неправильных $\tilde{A}$ чисел, которые определяются системой контроля как правильные
418, 419	420, 421, 422, 423, 424, 425, 426, 427, 428

Показатель для количественной оценки достоверности контроля данных в КВ может быть представлен в виде соотношения $P_{дк} = V_{пс} / V_0$, где: $V_{пс} = M = \prod_{i=1}^{n} m_i$ – количество правильных, лежащих в информационном числовом [0, M] диапазоне, кодовых слов для данного КВ; $V_0 = V_{пс} + \Delta$ – общее количество кодовых слов, которые в результате проведения контроля данных считаются правильными; $\Delta = (N \cdot m_{n+1} - M)$ – количество правильных кодовых слов, которые в результате проведения контроля данных считаются правильными ($N =]M/m_{n+1}[$). В этом случае показатель достоверности определяется соотношением

$$P_{дк} = \frac{V_{пс}}{V_0} = \frac{V_{пс}}{V_{пс} + \Delta} = \frac{M}{M + N \cdot m_{n+1} - M} = \frac{M}{N \cdot m_{n+1}}.$$

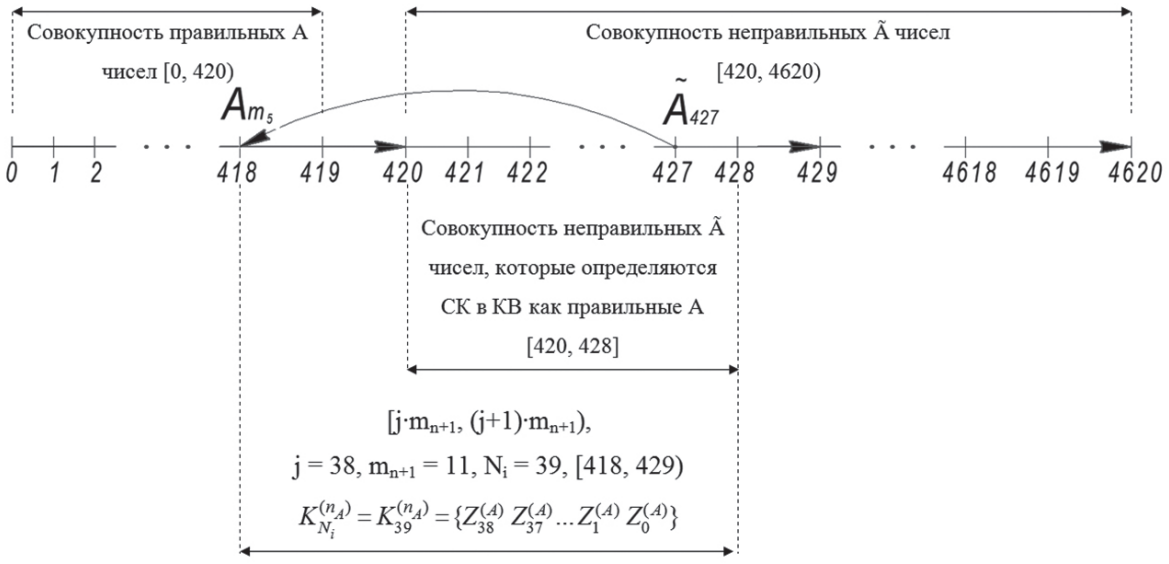


Рис. 3. Пример реализации операции контроля данных в КВ для $m_{n+1} = m_5 = 11$

Очевидно, что при $\Delta = 0$ показатель достоверности максимальный и равен $P_{\text{дк}} = 1$ ($M = N \cdot m_{n+1}$) .

Чтобы обеспечить максимальную достоверность контроля ($P_{\text{дк}} = 1$) данных для выражения $N_i =]M/m_i[$ необходимо выбрать модуль m_i, определяющий номер j числового интервала $[j \cdot m_i, (j+1) \cdot m_i)$ нахождения числа $A=(a_1, a_2, ..., a_i, ..., a_n, a_{n+1})$, из совокупности информационных n модулей КВ, кратных значению M. В этом случае $\alpha = M - [M / m_i] \cdot m_i = 0$ (где $[M/m_i]$ – целая часть числа M/m_i его не большая), что и обеспечивает максимальное значение показателя достоверности контроля $P_{\text{дк}} = 1$.

Пример 4. Из вышеприведенного КВ выбираем информационное основание $m_i = m_1 = 3$. При этом $N_i = N_1 = M/m_1 = = 4 \cdot 5 \cdot 7 = 140$. В этом случае рабочий числовой $[0, M_0)$ диапазон КВ разбивается на интервалы $[j \cdot m_i, (j+1) \cdot m_i)$, т.е. $[j \cdot m_1, (j+1) \cdot m_1)$. Для значения $m_1 = 3$ информационный интервал $[0, M)$ разбивается на $N_1 = M/m_1 =$ $= 140$ отрезков длиною три единицы (см. рис. 4). В таблице приведено содержимое БКН относительно основания $m_1 = 3$.

Пусть необходимо провести контроль числа $A=(01,11,010,000,1001)$. По значению $a_1 = 01$ в БКН (табл. 3) выбираем константу $KH_{m_1}^{(A)} = (01,01,001,001,0001)$.

Таблица 3

Содержимое БКН для $m_1 = 3$

a_i	Константы				
	$m_1 = 3$	$m_2 = 4$	$m_3 = 5$	$m_4 = 7$	$m_5 = 11$
00	00	00	000	000	0000
01	01	01	001	001	0001
10	10	10	010	010	0010

Далее определяем

$$A_{m_1} = A - KH_{m_1}^{(A)} = (00,10,001,110,1000).$$

Если $A_{m_1} - n_A \cdot m_1 = 426 - 142 \cdot 3 = 0$, то ОК имеет вид $K_{N_i}^{(n_A)} = K_{140}^{(142)} = \{Z_{139}^{(A)}\ Z_{138}^{(A)} \ldots Z_1^{(A)}\ Z_0^{(A)}\} = \{11...11...11\}$. Так как $N_i = 140 < n_A = 142$, то есть ошибка в числе A.

Проверка: $A = 427 > M = 420$. Число $A > M$, т.е. оно неправильное (искажено).

Отметим, что количество оборудования СК в основном зависит от количества сумматоров, реализующих операции вида (1). Таким образом, количество оборудования СК зависит от значения N_i ($i = \overline{1,n}$). В этом случае для минимизации количества оборудования СК в КВ необходимо выбрать максимальный по величине информационный модуль. Для упорядоченного ($m_i < m_{i+1}$) КВ это будет основание m_n.

Предварительная оценка количества оборудования для l-байтового машинного слова СПОД может быть проведена посредством коэффициента эффективности представленного в виде:

$$K_{\text{эф}}^{(l)} = \frac{N_1}{N_n} = \frac{M / m_1}{M / m_n} = \frac{M_1}{M_n} = \frac{m_n}{m_1}.$$

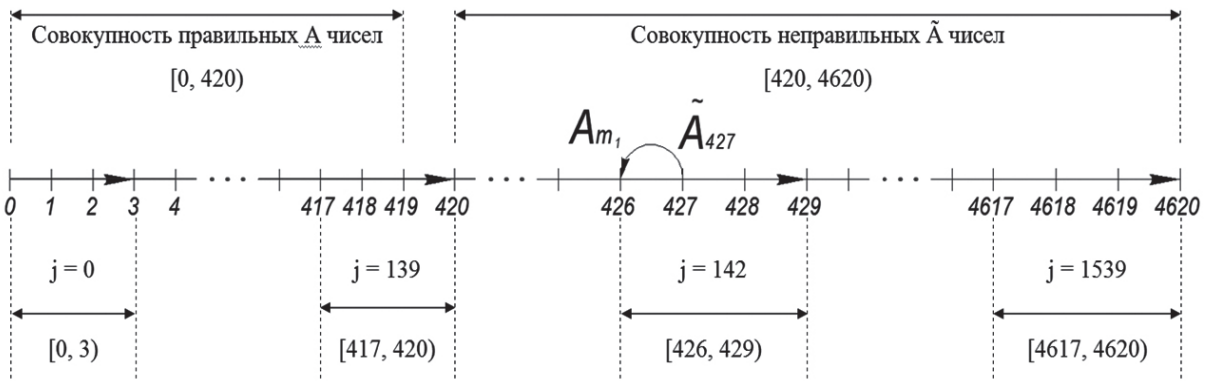
Пример 5. Максимальным из информационных оснований данного КВ является $m_n = m_4 = 7$. При этом $N_i = N_4 = M/m_4 = 3 \cdot 4 \cdot 5 = 60$. Рабочий числовой $[0, M_0)$ диапазон разбивается на интервалы $[j \cdot m_4, (j+1) \cdot m_4)$, т.е. на $M_0/m_4 = 4620/7 = 660$ отрезков. Для значения $m_4 = 7$ информационный $[0, M)$ интервал разбивается на $N_4 = M/m_4 = 60$ числовых отрезков длинною семь единиц (см. рис. 5). В табл. 4 приведено содержимое БКН относительно основания $m_4 = 7$ (табл. 4).

Таблица 4

Содержимое БКН для $m_4 = 7$

a_4	Константы				
	m_1=3	m_2=4	m_3=5	m_4=7	m_5=11
000	00	00	000	0000	0000
001	01	01	001	001	0001
010	10	10	010	010	0010
011	00	11	011	011	0011
100	01	00	100	100	0100
101	10	01	000	101	0101
110	11	10	001	110	0110

Пусть необходимо провести контроль числа $A=(01,11,010,000,1001)$. По значению a_4=000 в БКН (табл. 4) выбираем константу



$$j = 142,\ m_1 = 3,\ N_i = 140$$

$$K_{N_i}^{(n_A)} = K_{140}^{(142)} = \{Z_{139}^{(A)}\ Z_{138}^{(A)} \ldots Z_1^{(A)}\ Z_0^{(A)}\}$$

Рис. 4. Пример реализации операции контроля данных в КВ для $m_1 = 3$

$KH_{m_n}^{(A)} = KH_7^{(A)} = (00,00,000,000,0000)$. Далее определяем значение

$$A_{m_n} = A_7 = A - KH_7^{(A)} = (01,11,010,000,1001).$$

Посредством реализации соотношений (2) формулируем ОК

$$K_{N_4}^{(n_A)} = K_{60}^{(61)} = \left\{Z_{59}^{(A)}\, Z_{58}^{(A)} \ldots Z_1^{(A)}\, Z_0^{(A)}\right\} = \{11\ldots11\ldots11\}.$$

Исходя из вида ОК и используя выражение $A_{m_n} - n_A \cdot m_n = 0$, определяем, что n_A=61 ($A_{m_n} - n_A \cdot m_n = 427 - 61 \cdot 7 = 0$).

Так как $N_4 = 60 < n_A = 61$, то ошибка в данных A присутствует.

Проверка: $A = 427 > M = 420$.

В табл. 5 приведены расчетные данные сравнительного анализа оперативности контроля и выигрыша в сокращении количества оборудования системы контроля в КВ в относительных единицах.

Выводы. Таким образом, в статье разработан метод контроля данных в КВ, который в отличие от известных, основан на использовании ППНК n_A, что повышает оперативность контроля данных в СПОД. Использование табличного метода реализации арифметических операций позволило сформировать ОК, посредством которого всего за три условных временных такта получить ППНК n_A. В этом случае метод оперативного контроля данных в КВ, включающий алгоритм формирования n_A, сводиться к простой реализации операции сравнения чисел n_A и N_i (определение местоположения нулевого разряда $Z_{n_A}^{(A)}$ в записи ОК $K_{N_i}^{(n_A)}$, что технически можно реализовать за один машинный такт, путем использования одного многовходового элемента И). Если $n_A \geq N_i$ (все двоичные разряды ОК единичны), то число А считается неправильным, а если $n_A < N_i$ (в записи ОК присутствует один нулевой разряд $Z_{n_A}^{(A)}$), то число A – правильное.

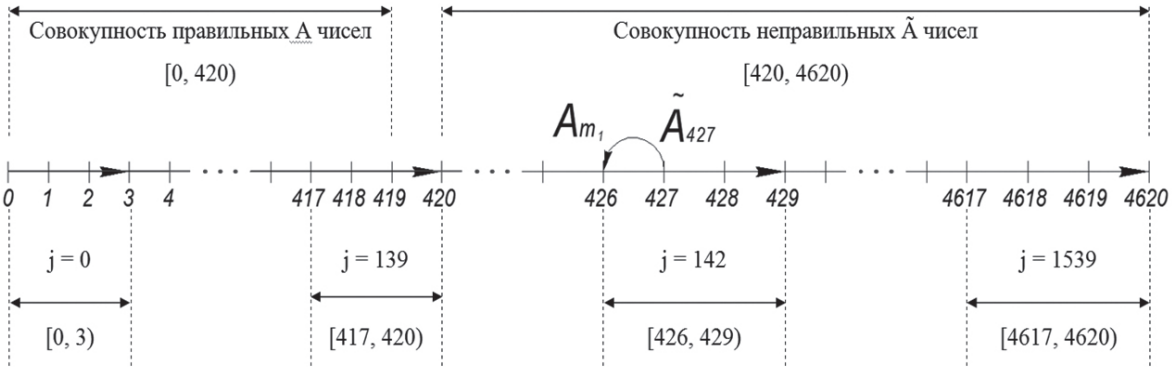
Расчетные данные и сравнительный анализ оперативности контроля данных по времени обнаружения ошибок и количества оборудования СК в КВ (табл. 5) показал, что с ростом разрядной сетки обрабатываемых данных в СПОД, что характерно для современной тенденции развития информационно-телекоммуникационных систем, эффективность непозиционного кодирования в классе вычетов существенно возрастает.

Литература

[1] *И.Я. Акушский, Д.И. Юдицкий*. Машинная арифметика в остаточных классах. – М.: Советское радио, 1968. – 440 с.

[2] *S.A. Koshman, V.I. Barsov, V.A. Krasnobayev, K.V. Yaskova, N.S. Derenko*. Method of bit-by-bit tabular realization of arithmetic operations in the system of residual classes // Радіоелектронні і комп'ютерні системи. – 2009. – № 5 (39). – С. 44–48.

[3] *Мороз С.А., Краснобаев В.А.* Исследование путей повышения эффективности использования информационно-телекомуникационных систем на основе применения непозиционных кодовых структур класса вычетов // Системи озброєння та військова техніка: Науковий журнал. – Х.: ХУПС ім. Івана Кожедуба. – 20011. – № 1 (25). – С. 114-118.



$j = 142,\ m_1 = 3,\ N_i = 140$

$$K_{N_i}^{(n_A)} = K_{140}^{(142)} = \{Z_{139}^{(A)}\, Z_{138}^{(A)} \ldots Z_1^{(A)}\, Z_0^{(A)}\}$$

Рис. 5. Пример реализации операции контроля данных в КВ для $m_4 = 7$

Таблица 5

Сравнительные данные оперативности контроля данных и количества оборудования СК в КВ

Длина $l(n)$ разрядной сетки СПОД	Условное время контроля данных			Выигрыш в сокращении количества оборудования СК		
	Наилучший из известных методов контроля в КВ	Разработанный метод контроля (ППНК)	Выигрыш во времени контроля данных в [%]	m_1	m_n	$K_{эф}^{(l)}$
1 (4)	8	4	50	3	7	2
2 (6)	12	4	66	2	13	6
3 (8)	16	4	75	3	19	6
4 (10)	20	4	80	2	29	14,5
8 (16)	32	4	88	2	53	26,5

[4] *С. А. Мороз, В. А. Краснобаев.* Метод контроля информации в непозиционной системе счисления класса вычетов // Системи управління, навігації та зв'язку. – 2011. – Вип. 2 (18). – С. 134-138.

[5] *Мороз С. А., Краснобаев В.А.* Метод контроля данных, представленных кодом непозиционной системы счисления класса вычетов // Радиоэлектроника и информатика. 2011. Вып. № 1 (52). С. 47-51.

Поступила в редколлегию 14.03.2012

Мороз Сергей Александрович, аспирант кафедры автоматизации и компьютерно-интегрированных технологий Харьковского национального технического университете сельского хозяйства им. Петра Василенка. Область научных интересов: создание отказоустойчивого спецпроцессора быстрой и достоверной обработки данных на основе использования непозиционной системы счисления в классе вычетов.

Краснобаев Виктор Анатольевич, профессор кафедры автоматизации и компьютерных технологий Харьковского национального технического университета сельского хозяйства им. Петра Василенко, доктор техн. наук, профессор, Заслуженный изобретатель Украины, Почётный радист СССР. Область научных интересов: теоретическое обоснование и практическое создание сверхбыстродействующих и высокоотказоустойчивых вычислительных структур в модулярной арифметике.

Замула Александр Андреевич, фото и сведения об авторе см. на с. 193.

УДК 681.142:681.3

Метод оперативного контролю даних в класі вирахувань на основі використання позиційної ознаки непозиційної коди / С.А. Мороз, В.А. Краснобаєв, О.А. Замула // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 281–287.

У даній статті розглядається метод оперативного контролю даних, представлених кодом класу вирахувань. Наведені приклади конкретного виконання операцій контролю.

Ключові слова: немодульні (позиційні) операції, кодова структура, контроль даних, непозиційна система числення.

Табл. 05. Іл.05. Бібліогр.: 05 найм.

UDC 681.142:681.3

Method of operative data control in the residue class on the basis of using a nonpositional code position sign / S.A. Moroz, V.A. Krasnobaev, A.A. Zamula // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 281–287.

The method of operative control of data represented by a residue class code is considered in the paper. Specific examples of control operations are given.

Keywords: nonmodular (position) operations, code structure, data control, nonpositional system of notation.

Tab. 05. Fig. 05. Ref.: 05 items.

УДК 004.652: 004.655: 004.432.4

ВАРИАНТЫ ИСПОЛЬЗОВАНИЯ ОПЕРАТОРОВ ЯЗЫКА МОДЕЛИ ДАННЫХ

В.И. ЕСИН, М.В. ЕСИНА

Описывается проблема использования различных языковых средств по поддержанию ряда функций интероперабельных информационных систем, которая может быть решена с помощью специального языка модели данных. Приводятся возможные варианты использования операторов этого языка и даются некоторые рекомендации по их применению.

Ключевые слова: модель данных, универсальная модель данных, язык модели данных, база данных.

ВВЕДЕНИЕ

Анализ использования различных языковых средств [1-5] по поддержанию ряда функций интероперабельных информационных систем, связанных с представлением знаний о предметной области (необходимых для разработки формализованных описаний при концептуальном моделировании); созданием программных приложений; разработкой средств семантического согласования; разработкой средств разграничения прав доступа к данным и т. д., показал что они:

- как правило, ориентированы на поддержку только отдельных выше названных функций;
- требуют знания модели (схемы) базы данных или структуры источника данных;
- не в состоянии стать подходящим средством выражения семантики данных и способствовать решению задачи автоматической трансформации семантически правильного запроса, составленного в терминах предметной области (ПрО) с использованием естественного языка, в синтаксически и терминологически корректный запрос к конкретной базе данных;
- достаточно сложны в освоении не специалистами в области информационных технологий.

Все это послужило поводом для разработки специального языка модели данных, лишенного указанных выше недостатков. И таким языком стал язык модели данных (ЯМД) [6].

В работе [6] уже говорилось о ЯМД как о специальном непроцедурном (декларативном) основывающимся на метаонтологиях модели «объект-событие» языке, который позволяет определять метаданные и данные любой предметной области, а также манипулировать ими в терминологии близкой к естественному языку. Но операторы (точнее строки метаописания, составленные из операторов) ЯМД могут использоваться не только для описания элементов рассматриваемой ПрО и манипулирования данными, занесенными в базу данных (БД) с универсальной моделью данных (УМД). Они могут применяться и в других различных целях, способствующих эффективному и корректному использованию данных, хранящихся в различных источниках. Рассмотрим эти возможные варианты.

ИСПОЛЬЗОВАНИЕ ОПЕРАТОРОВ ЯМД

1. Операторы ЯМД используются в соответствующих информационных системах (ИС), когда необходимо реализовывать механизм распределения прав доступа к данным вплоть до любого конкретного элемента данных.

С этой целью в рамках ядра схемы БД с УМД была создана таблица «*Ограничения прав доступа*» к конкретному элементу данных. На псевдокоде ее можно описать следующим образом:

```
USER_ID INTEGER not null, -- Идентификатор
                пользователя
USER_NAME VARCHAR(255) not null -- Имя
                пользователя
CLASS_ID  INTEGER not null, -- ID-объекта
                в таблице TABLE_NAME
TABLE_NAME VARCHAR(255) not null, -- имя
                таблицы, для которой создается
                ограничение прав доступа
DESCRIPTOR CLOB not null -- строка
                метаописания ЯМД
primary key (CLASS_ID, TABLE_NAME) --
                PK-первичный ключ.
```

В данной таблице в соответствующем атрибуте (DESCRIPTOR) хранится строка метаописания языка модели данных. На основании значения этого атрибута определяется адрес защищаемого элемента (его идентификатор *ID* в требуемой таблице БД с УМД), доступ к которому ограничивается. Процесс заполнения атрибутов данной таблицы автоматизирован с помощью специального программного приложения, использующего в качестве основного входного параметра именно ту строку метаописания, которая задает адрес защищаемого элемента данных.

Поясним данный механизм на примере ограничения прав доступа пользователя «*AVT1*» к экземпляру объекта «*333333*» класса «*Техническое средство*» и типа «*Вольво*», принадлежащего пользователю «*AVT2*».

Вначале составляется строка метаописания, по которой однозначно можно определить адрес данного экземпляра объекта в БД с УМД. В нашем случае вид этой строки будет следующий:

{<*Раздел*>=Хар_УГЭС**; /<*КлассО*>**=Техническое средство**; <*ТипО*>**=*Вольво*; <*ЭкзО*>**=*333333*;}.

Затем запускается на выполнение специальная программа определения прав доступа к данным. При этом подключение в ней к БД с УМД

должно осуществляться с правами того пользователя, данные которого он хочет ограничить в использовании другому пользователю (рис. 1).

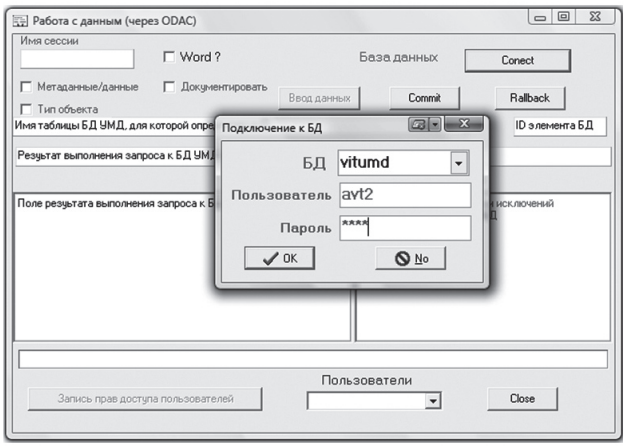


Рис. 1. Программа задания прав доступа к данным

В нашем случае пользователь «*AVT2*» ограничивает использование экземпляра объекта «*333333*» для пользователя «*AVT1*». Для этого он (пользователь «*AVT2*») с помощью вышеназванной программы выбирает в окне «Пользователи» пользователя «*AVT1*» и исполняет приведенную выше строку метаописания ЯМД с фиксацией «кнопки» «*Запись прав доступа пользователей*» (рис. 2).

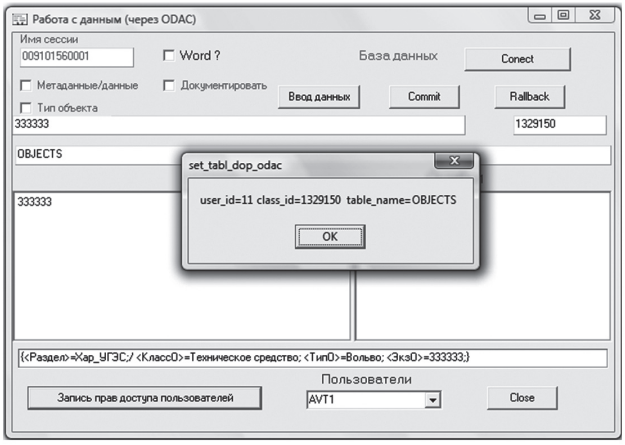


Рис. 2. Ограничение прав доступа пользователя AVT1

После чего данные полей приложения: «*AVT1*», «*OBJECT*», «*1329150*», {<***Раздел***>=Хар_УГЭС;/ <***КлассО***>=Техническое средство; <***ТипО***>=Вольво; <***ЭкзО***>=333333;} автоматически записываются в таблицу ограничения прав доступа, в соответствующий ее атрибут. Просмотр содержимого таблицы с помощью запроса

```
SELECT * FROM "Ограничения прав доступа"
WHERE CLASS_ID=1329150
```

покажет следующий результат:

USER_ID	CLASS_ID	TABLE_NAME	DESCRIPTOR	USER_NAME
11	1329150	OBJECTS	{<Раздел>=Хар_УГЭС;/ <КлассО>=Техническое средство; <ТипО>=Вольво; <ЭкзО>=333333;}	AVT1

На основе данных этой таблицы, реализованные в схеме БД с УМД политики (речь о которых шла в [7]) автоматически обеспечивают защиту соответствующих данных от несанкционированного использования. Теперь если пользователь «*AVT1*» захочет обратиться из любого приложения (будь то Oracle SQL*Plus, PL/SQL Developer либо любое другое) к данным БД с УМД, то экземпляр объекта «*333333*» ему будет не доступен, так как доступ к нему «закрыт» на уровне сервера СУБД.

В подтверждение сказанному ниже приведены обращения пользователей «*AVT1*» и «*AVT2*» к БД с УМД, выполненные с помощью различных приложений. Так на рис. 3, 4 приведены данные, выводимые специальным приложением программного инструментария разработчика БД с УМД – программой просмотра, для пользователей «*AVT1*» и «*AVT2*» соответственно.

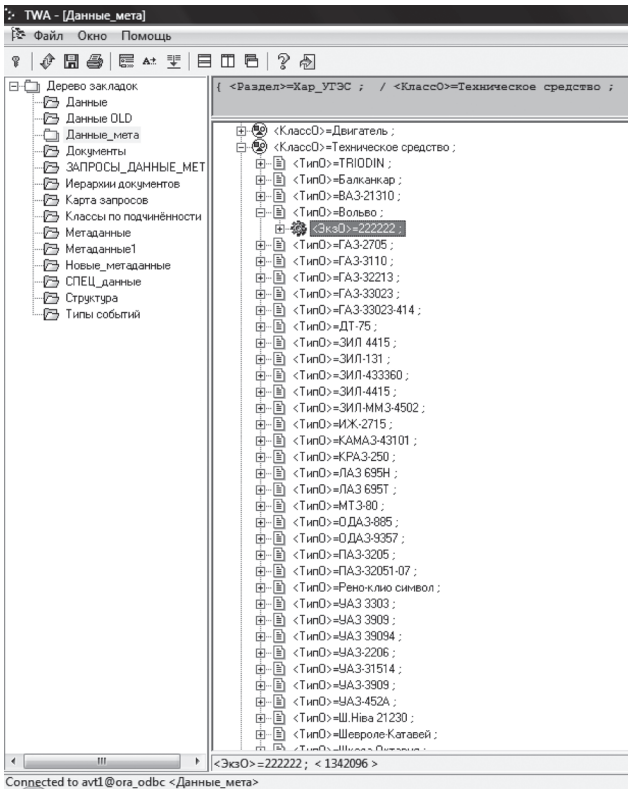


Рис. 3. Данные, выводимые программой просмотра для пользователя AVT1

Как видно из этих рисунков, пользователю «*AVT1*», в отличие от пользователя «*AVT2*», не доступен экземпляр объекта «*333333*».

При обращении к данным в среде PL/SQL Developer с помощью следующего кода:

```
declare
id number;
pr_type number;
text_vx_str varchar2(5000);
tmp_char varchar2(2000);
cur_name_end_text data_umd_big.my_ref_cursor;
i_count number;
i number;
name_session_proc varchar2(255);
```

```
text_exit varchar2(250);
error_text varchar2(5000);
pr_write number;
begin
pr_type:=1; id:=0; i_count:=0; pr_write:=0;
      text_vx_str:='{<Раздел>=Хар_
      УГЭС;/<КлассО>=Техническое средство;
      <ТипО>=Вольво; <ЭкзО>=*.*?;}';
DATA_UMD_BIG_P(text_vx_str,  pr_type,  pr_
write, id, name_session_proc, text_exit, er-
ror_text, cur_name_end_text);
i:=0;
LOOP
      fetch cur_name_end_text into tmp_char;
      exit when cur_name_end_text%NOTFOUND;
      i:=i+1;
dbms_output.put_line('name'||'('||i||')='||tmp_
char);
      END LOOP;
end;
```

результат для пользователя «*AVT2*» будет:

```
name(1)=222222
name(2)=333333,
```

а для пользователя «*AVT1*»:

```
name(1)=222222.
```

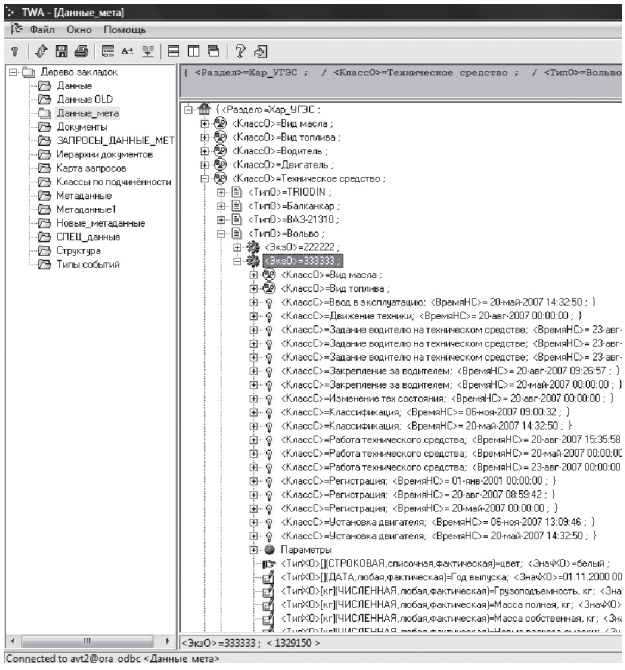


Рис. 4. Данные, выводимые программой просмотра для пользователя AVT2

Результат обращения пользователя «*AVT1*» к БД с УМД в среде Oracle SQL*Plus приведен на рис. 5.

Как видим результат один и тот же. Пользователь «*AVT1*» не имеет доступа к экземпляру объекта «*333333*». Если даже обратиться напрямую к таблице экземпляров объектов посредством оператора SELECT языка запросов SQL, результат для пользователя «*AVT1*» будет таким же – объект «*333333*» ему недоступен.

Аналогичным образом можно ограничить доступ различным пользователям к любым метаданным и данным любой предметной области.

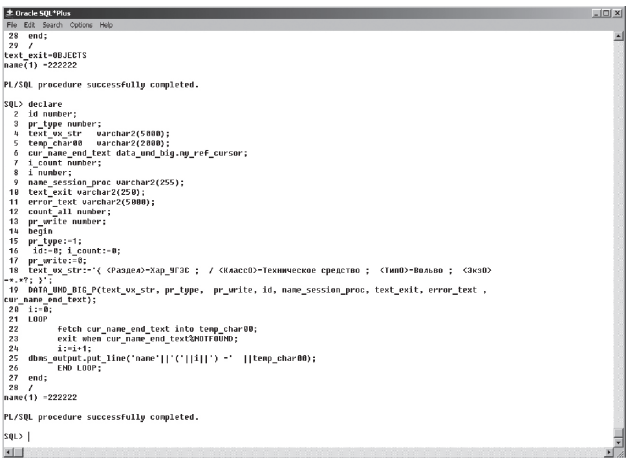


Рис. 5. Результат исполнения кода для пользователя AVT1 в среде Oracle SQL*Plus

2. Сегодня можно говорить, что эра, когда разработчики информационных систем приходили в организацию и начинали проекты информатизации «с нуля», прошла. Большинство организаций уже имеет некоторые информационные системы, которые со временем становятся бременем компании и начинают требовать реинжиниринга. Это объясняется тем, что специалистам любой крупной и давно существующей компании, которая обладает несколькими базами данных, относящимися к разным видам деятельности, данные в которых могут иметь разные представления и быть даже несогласованными, становится очень трудно связывать и анализировать содержащуюся в них информацию. К тому же многие базы данных, составляющие основу таких ИС, как правило, построены на уже «устаревших платформах» (например, Dbase, FoxPro и т. д.). При этом данные, в них хранящиеся, имеют большую практическую ценность.

Как результат, имея самые разнообразные изолированные друг от друга источники данных, найти необходимые для деятельности компании данные становится невозможным. Поэтому чтобы упростить и ускорить доступ к такой информации, дать возможность прослеживать связи между разными источниками данных и обеспечить надежность использования унаследованных настольных баз данных в обеспечении управления на всех уровнях, необходимо каким-то образом интегрировать эту информацию.

Сегодня организации обычно расходуют от 20 до 40% своего IT-бюджета на эволюционирование своих данных путем миграции (изменение местоположения данных), преобразования (изменения формы или структуры данных) или очистки (изменение или повторный ввод данных для последующего использования) [8].

А раз подобные задачи встраивания устаревших информационных компонентов в систему, основанную на новой технологии, сегодня достаточно часто приходиться решать, то нужно сделать так, чтобы они были разрешимыми. То есть, чтобы компоненты унаследованных систем

сохраняли интероперабельность. Потому что, чем дольше живет и приносит пользу информационная система, тем это выгоднее для компании. Естественно, что для этого должна существовать возможность добавления в нее компонентов, спроектированных и разработанных, вообще говоря, в другой технологии. Сегодня в мире существует большое количество подходов, методов и технологических решений, напрямую или косвенно соотносимых с деятельностью по реинжинирингу ИС. Однако они не интегрированы на уровне методологий (процессов разработки). Как результат, можно наблюдать наличие огромного их количества, где основной акцент сделан на опять же разработку ИС «с нуля», и практическое отсутствие «стройных» методологий, целью создания которых являлось бы комплексное, целостное решение задач реинжиниринга ИС [9].

С этой целью в свое время был предложен метод решения этой проблемы [10], который в основе своей опирается как раз на использование ЯМД. Строки метаописания ЯМД используются в нем при конвертации данных из БД других информационных систем, построенных на различных платформах (таких как Oracle, PostgreSQL, Access), в БД с УМД. Более подробно о данном механизме излагалось в [10].

3. Строки метаописания ЯМД также могут быть использованы при создании отчетных документов, формирующихся, с помощью различных средств, в том числе и OLAP-технологии.

На рис. 6 приведен интерфейс приложения, использующего строки метаописаний ЯМД, для создания отчетных документов в форматах: PDF, RTF, HTML, XLS, CSV, XML, Jasper Reports, JRPXML.

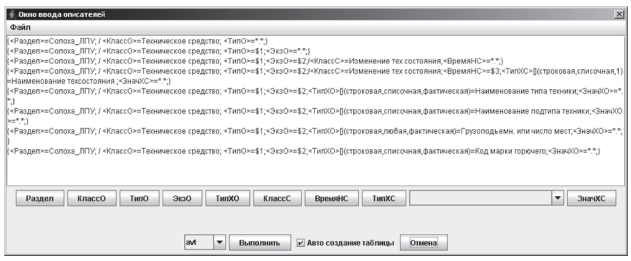


Рис. 6. Интерфейс приложения для формирования отчетных документов

Формы некоторых отчетных документов, полученные с использованием возможностей данного приложения, приведены на рис. 7, 8.

4. Сегодня для балансировки нагрузки на телекоммуникационные и вычислительные ресурсы распределенной системы очень часто используют механизм репликаций. И поэтому, чтобы обеспечить наиболее простую организацию этого механизма, при использовании баз данных с УМД вместо моментальных снимков основных таблиц (*snapshot*) БД асинхронно распространяется (реплицируется) таблица-журнал измененных данных (табл. 1). Из которой впоследствии извлекаются и выполняются строки метаописаний ЯМД, что позволяет существенно снизить нагрузку на систему и ее трафик.

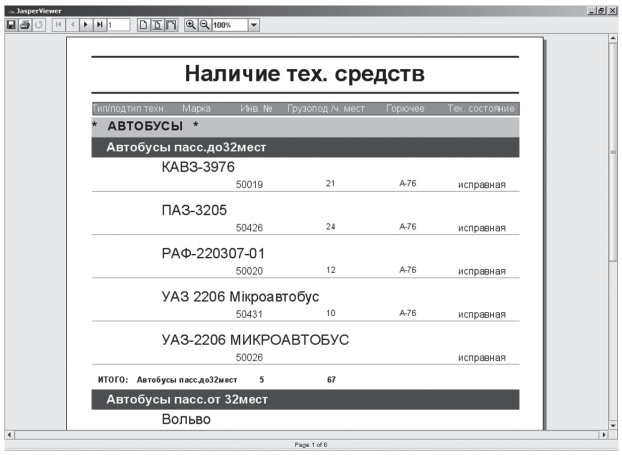


Рис. 7. Форма отчетного документа в Jasper Viewer

Рис. 8. Диаграмма распределения технических средств по типам

Таблица 1

Пример таблицы-журнала измененных данных

Системное имя	IP-адрес компьютера	Имя пользоват.	Имя БД	Строка метаописания ЯМД	Данные или метаданные	Операция*	Время записи в текущую БД	Время выдачи данных в БД распределен. системы
VIT\vitaly	195.110.100.1	SHEMA_1	Ora_umd	{<Раздел>=Нов_1;}	метаданные	insert	16.09.2008 10:00:01	17.09.2008 20:10:11
Nik\mycmp	101.63.102.28	SHEMA_2	spumd	{<Раздел>=Хар_УГЭС;/ <КлассО>=Тех. средство; <ТипО>=Вольво; <ЭкзО>=111; <УдалитьО>=;}	данные	delete	26.07.2010 17:41:48	29.07.2010 17:41:48

* – операция чтения (*select*) не отслеживается.

При этом автоматически решается проблема целостности транзакций: внесение изменений в

реплицируемую БД с УМД осуществляется в ней же самой, путем исполнения переданных строк метаописаний ЯМД.

ВЫВОДЫ

1. Рассмотренные различные варианты использования строк метаописания ЯМД способствуют эффективному и корректному использованию (в соответствие с распределением прав доступа вплоть до любого конкретного элемента) данных, хранящихся в различных источниках.

2. Для того чтобы предложенные варианты использования строк метаописания ЯМД стали на самом деле эффективным средством в руках различных пользователей, последние должны иметь (сформировать) правильное представление о ПрО, которое затем следует адекватно отобразить в схеме БД с УМД. Независимо от того было получено это отображение в процессе первоначального моделирования базы данных или при переносе ее из других источников информации в схему БД с УМД.

Литература

[1] *Калиниченко Л. А.* СИНТЕЗ: язык определения, проектирования и программирования интероперабельных сред неоднородных информационных ресурсов / Л. А. Калиниченко. – М. : ИПИ РАН, 1993. – 121 с.

[2] *Шрайбман В.* Выражение семантики данных. RDF против XML [Электронный ресурс] / В. Шрайбман. – Режим доступа : http://citforum.ru/internet/xml/rdf_xml/

[3] RDF схема метаданных ИСИР / [Бездушный А. А., Бездушный А. Н., Жижченко А. Б. и др.] // Современные технологии в информационном обеспечении науки – М. : Научный Мир, 2003. – С.141–159.

[4] *Токмаков Д. И.* Использование средств языка RDF в аннотировании Интернет-ресурсов / Д. И. Токмаков // Информационные ресурсы России. – М. : Федеральное государственное учреждение Российское энергетическое агентство Министерства энергетики Российской Федерации, 2007. – № 5. – С.30-33.

[5] RDF Vocabulary Description Language 1.0: RDF Schema W3C Recommendation. [Electronic resource] – Access mode : http://www.w3.org/TR/rdf-schema/

[6] *Есин В. И.* Язык для универсальной модели данных / В. И. Есин, М. В. Есина // Системи обробки інформації. – Х. : Харківський університет Повітряних Сил, 2011. – № 5(95) – С.193–197.

[7] *Есин В. И.* Особенности защиты данных в базах данных с универсальной моделью / В. И. Есин, М. В. Есина // Прикладная радиоэлектроника. – Х. : Харьковский национальный университет радиоэлектроники, 2011. – Т. 10, № 2 – С.226–232.

[8] *Кузнецов С.* Модель зрелости для управления данными [Электронный ресурс] / С. Кузнецов – Режим доступа : http://www.citforum.ru/computer/2007-04/

[9] *Ахтырченко К. В.* Методы и технологии реинжиниринга ИС / К. В. Ахтырченко, Т. П. Сорокваша // Труды Института системного программирования РАН. / Под редакцией чл.-корр. РАН В. П. Иванникова. – М. : ИСП РАН, 2003. – Т. 4. – С.141–162.

[10] *Сорока Л.С.* Реинжиниринг существующих баз данных в базу с универсальной моделью данных и возможностью распараллеливания процессов / Л. С. Сорока, В. И. Есин, М. В. Есина // Сборник научных трудов Параллельная компьютерная алгебра: Всероссийская научная конференция с элементами научной школы для молодежи, г. Ставрополь, 11-15 октября 2010 г. Ставропольский государственный университет. – Ставрополь: Издательско-информационный центр «Фабула», 2010, С.298-304.

Поступила в редколлегию 29.03.2012

Есин Виталий Иванович, кандидат технических наук, доцент, доцент кафедры безопасности информационных систем и технологий ХНУ им. В.Н. Каразина. Область научных интересов: модели и методы разработки баз данных информационных систем и обеспечение их безопасности.

Есина Марина Витальевна, студентка факультета компьютерных наук ХНУ им. В.Н. Каразина. Область научных интересов: базы данных и обеспечение их безопасности.

УДК 004.652: 004.655: 004.432.4

Варіанти використання операторів мови моделі даних / В.І. Єсін, М.В. Єсіна // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 288–292.

Описується проблема використання різних мовних засобів по підтримці ряду функцій інтероперабельних інформаційних систем, яка може бути вирішена за допомогою спеціальної мови моделі даних. Наводяться можливі варіанти використання операторів цієї мови і даються деякі рекомендації по їх вживанню.

Ключові слова: модель даних, універсальна модель даних, мова моделі даних, база даних.

Табл. 01. Іл.08. Бібліогр.: 10 найм.

UDC 004.652: 004.655: 004.432.4

Variants of using data model language operators / V.I. Yesin, M.V. Yesina // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 288–292.

The problem of using different language means of supporting some functions of interoperable informative systems is described. This problem can be solved by means of a special data model language. Possible variants of using operators of this language and some recommendations for their use are given.

Keywords: data model, universal data model, data model language, database.

Tab. 01. Fig. 08. Ref.: 10 items.

УДК 621.391.1

СИНТЕЗ СИСТЕМ СИГНАЛОВ С ЗАДАННЫМИ КОРРЕЛЯЦИОННЫМИ СВОЙСТВАМИ, ЗАКОНАМИ ФОРМИРОВАНИЯ, СТРУКТУРНЫМИ И АНСАМБЛЕВЫМИ СВОЙСТВАМИ

И.Д. ГОРБЕНКО, А.А. ЗАМУЛА

Рассматривается задача синтеза дискретных сигналов с заданными корреляционными, структурными и ансамблевыми свойствами.

Ключевые слова: ансамбль сложных сигналов, помехоустойчивость, минимаксный критерий, синтез системы сигналов.

ВВЕДЕНИЕ

К системам передачи информации предъявляются все более жесткие требования по обеспечению работы в условиях сложных внешних воздействий: естественных и преднамеренных помех, помех от других радиотехнических систем, функционирующих на близких частотах или в общем участке диапазона частот.

Важными характеристиками некоторых систем передачи информации являются помехоустойчивость и скрытность функционирования. Под помехоустойчивостью понимают способность системы противостоять воздействию мощных помех. Скрытность функционирования системы предполагает способность системы функционировать в режиме, затрудняющим обнаружение передаваемых сообщений и оценку их параметров разведывательной аппаратурой злоумышленника. Одним из видов скрытности является информационная скрытность. Такой вид скрытности предполагает использование целого комплекса мер, методов и средств для затруднения определения злоумышленником: самого факта передачи сообщений по каналам связи; содержания передаваемых сообщений и др. Большое значение при решении задач обеспечения требуемой помехоустойчивости и скрытности функционирования (в том числе, информационной скрытности) имеют исследования, связанные с использованием новых видов сигналов, получивших название сложных, широкополосных, многомерных и шумоподобных. Разработка методов синтеза сложных сигналов с хорошими корреляционными, ансамблевыми, статистическими, структурными и другими свойствами является актуальной задачей.

Комплексное решение проблемы обеспечения помехоустойчивости, скрытности функционирования системы передачи информации может быть достигнуто, в том числе, на основе реализации динамического режима передачи информации, при котором соответствие: бит сообщения – сигнал меняется с течением времени по закону, предсказание которого возможно с вероятностью, не превышающей допустимую. Одним из путей достижения заданной помехоустойчивости, является реализация частотной избыточности в канале связи.

При радиоэлектронном противодействии эффективная помеха может быть организована только после обнаружения присутствия противостоящей системы в эфире и оценки таких ее параметров как частотный диапазон и занимаемая полоса. Если скрытная система использует сигнал с некоторым законом модуляции, параметры которого неизвестны перехватчику, то последний лишен возможности применения согласованного фильтра или коррелятора для обнаружения сигнала. В этих условиях у противостоящей системы нет иного выбора, как рассматривать перехватываемый сигнал в виде случайного и основывать его обнаружение только на факте появления или отсутствия некоторого избытка энергии в некотором участке частотного диапазона. Перехватчик применяет энергетический детектор, называемый также радиометром, который является оптимальным с точки зрения обнаружения ограниченного по полосе шумового сигнала на фоне аддитивного белого гауссовского шума [1].

Перехватчику могут быть неизвестны заранее сведения о частотном диапазоне и интервале времени, занимаемом сигналом. Учитывая эти обстоятельства, его стратегия будет заключаться в комбинировании указанных параметров, осуществляя процедуру обнаружения либо путем сканирования частотно-временной области, либо используя набор параллельных каналов, каждый из которых ответственен за анализ ограниченного участка частотно-временной области. В любом случае качество работы приемника системы-перехватчика будет полностью определяться характеристикой энергетического детектора, настроенного на истинную для перехватываемого сигнала частотно-временную зону. В свою очередь, у скрытной системы имеется только единственная возможность предотвратить обнаружение своего сигнала потенциальным перехватчиком: использовать сигналы с распределенным спектром, обладающие максимально возможным значением выигрыша от обработки (произведение полосы частот, занимаемой сигналом на его длительность). Единственной причиной, вынуждающей перехватчик прибегнуть к такому неэффективному инструменту как энергетический приемник, является

отсутствие информации о структуре обнаруживаемого сигнала, т.е. его закона модуляции. По этой причине перехватчик не может обрабатывать сигнал аналогично приемнику скрытной системы (т.е. осуществлять согласованную фильтрацию). Очевидно, что в случае недостаточной структурной сложности (скрытности) сигнала и осведомленности перехватчика о его возможных альтернативных вариантах, перехватчик может попытаться их все реализовать. Соответствующим оборудованием для этого может служить набор параллельных согласованных фильтров либо единый перестраиваемый фильтр (несколько фильтров), пригодный для обработки сигналов различных по структуре последовательно во времени. Поэтому другая сторона стратегии скрытной системы в борьбе с перехватчиком состоит в применении сигналов с практически не раскрываемой структурой. В качестве показателя оценки структурной скрытности сигнала, может быть использован следующий

$$S = \frac{l}{L}, \qquad (1)$$

где: L – период сигнала; l – количество символов сигнала, которое необходимо знать для формирования L-l оставшихся.

ОСНОВНОЕ СОДЕРЖАНИЕ ИССЛЕДОВАНИЙ

Усилия исследователей направлены на поиски ансамблей сложных сигналов, характеристики которых с ростом длины приближаются к границе «плотной упаковки» [2], т.е. ансамбля, все представители которого обладают нулевой постоянной составляющей, идеальной периодической автокорреляционной функцией (ПФАК), нулевой периодической функцией взаимной корреляции (ПФВК), заданным объемом системы сигналов. Широко распространенным критерием подобного приближения является минимаксный критерий, ориентирующий синтез ансамбля на минимизацию максимального значения на множестве всех нежелательных корреляций. Для идеального ансамбля корреляционный пик как наибольшее из двух величин: максимума среди всех боковых лепестков автокорреляций последовательностей и максимума среди значений взаимных корреляций всех пар последовательностей равны нулю, а для любого реального ансамбля корреляционный пик может служить адекватной мерой его близости к идеальному.

Ансамбли со значением корреляционного пика достигающие предела, предсказываемого нижними границами Велча и Сидельникова [1], являются оптимальными по критерию корреляционного пика, и иногда называются минимаксными.

Синтез семейств сигналов с необходимыми авто и взаимно корреляционными свойствами заключается в отыскании семейства дискретных последовательностей, обладающего соответствующими авто и взаимно корреляционными функциями.

Обсуждаются методы синтеза оптимальных бинарных последовательностей большой длины с заданными авто-, взаимно- корреляционными и ансамблевыми свойствами.

К настоящему времени нет единой теории синтеза систем дискретных сигналов (ДС) с заданными авто-, взаимно-, стыковыми корреляционными свойствами. По существу, на основе комплексного использования аппарата теории полей Галуа, разностных множеств и комбинаторики, а также теории чисел, к сегодняшнему дню в основном развита теория анализа и синтеза двоичных линейных рекуррентных последовательностей максимального периода и линейных рекуррентных последовательностей (ЛРПТ) с одно- и трехуровневой ПФАК [3], а также характеристических дискретных сигналов (ХДС) с одно – и двухуровневой ПФАК [2]. Однако, как показали исследования, введение жестких ограничений на вид ПФАК ДС существенно ограничивает возможности источников сигналов с точки зрения улучшения ансамблевых и структурных свойств [3].

Сформулируем задачу синтеза одного класса сигналов с заданными корреляционными ансамблевыми и структурными свойствами, обеспечивающих требуемые значения помехозащищенности, имитостойкости и скрытности функционирования системы передачи информации. Потребуем, что бы такие системы сигналов обладали свойством «размытости» по корреляционным свойствам. Указанное свойство означает, что увеличение или уменьшение длины дискретной последовательности не изменяет корреляционные свойства, присущие исходной дискретной последовательности.

Под задачей синтеза сигналов будем понимать задачу построения словарей (подмножеств) векторов $(W_m^q), q = \overline{1,N}$, $m = \overline{1,M}$, вся $M_k << p^L$ совокупность которых образует систему сигналов размерности $M_k = N \times M_x$ таких, что в каждом из словарей выполняются следующие условия

1. Автосвертка или периодическая функция автокорреляции (ПФАК) каждого из W_m^q ДС удовлетворяет системе нелинейных параметрических неравенств (СНПН)

$$R_{a_1}^q(l) \le \sum_{i=1}^{L-1} W_i^q (W_{i+c}^q)^* \le R_{a_2}^q(l),$$

$$l = \overline{1, L-1}, \;\; q = \overline{1,N}, \qquad (2)$$

где $R_{a_1}^q(l)$ и $R_{a_2}^q(l)$ заданные (требуемые) реализации ПФАК;

2. Взаимная свертка (СФВК) (W^q W^p) ДС со стыковыми словами W^{qp} и W^{pq} удовлетворяет совокупности систем нелинейных параметрических неравенств:

$$R^{qp}_{b_{1,1}}(l) \le \sum_{i=0}^{L-K} W_i^q \times (W_{i+l}^p)^* + \sum_{i=L-K+1}^{L-1} W_i^q \times \times (W^p_{i-l+K})^* \le R^{qp}_{b_{2,1}}(l);$$

$$R^{qp}_{b_{1,2}}(l) \le \sum_{i=0}^{L-K} W_i^q \times (W_{i+l}^q)^* + \sum_{i=L-K+1}^{L-1} W_i^q \times \times (W^p_{i-l+K})^* \le R^{qp}_{b_{2,2}}(l);$$

$$R^{qp}_{b_{1,3}}(l) \le \sum_{i=0}^{L-K} W_i^q \times (W_{i+l}^p)^* + \sum_{i=L-K+1}^{L-1} W_i^q \times \times (W^q_{i-l+K})^* \le R^{qp}_{b_{2,3}}(l);$$

$$R^{qp}_{b_{1,4}}(l) \le \sum_{i=0}^{L-K} W_i^p \times (W_{i+l}^p)^* + \sum_{i=L-K+1}^{L-1} W_i^p \times \times (W^q_{i-l+K})^* \le R^{qp}_{b_{2,4}}(l);$$

$$R^{qp}_{b_{1,5}}(l) \le \sum_{i=0}^{L-K} W_i^p \times (W_{i+l}^q)^* + \sum_{i=L-K+1}^{L-1} W_i^p \times \times (W^p_{i-l+K})^* \le R^{qp}_{b_{2,5}}(l); \quad (3)$$

причем $l = \overline{1, L-1}$, для всевозможных сочетаний q и p, $q = \overline{1, N}$, $p = \overline{1, N}$, $q \ne p$, где $R^{qp}_{b_{1,j}}(l)$ и $R^{qp}_{b_{2,j}}(l)$ – заданные (требуемые) реализации ПФВК и СФВК.

3. Исследования показывают, что существенные затруднения в преодолении скрытности функционирования радиоканалов могут быть созданы за счет придания сигналам свойства «размытости». Введем понятие размытости. Причем вначале сформулируем задачу синтеза одиночного сигнала W^q, обладающего размытостью по циклической свертке. Определим интервал размытости Δx по длительности

$$L - x_2 \le \Delta x \le L + x_1, \quad (4)$$

Полагая, что в общем случае

$$|x_1| \ne |x_2|, |x_1|, |x_2| < L,$$

интервал размытости Δy относительно истинных значений цикловой частоты в виде

$$L - y_2 \le \Delta y \le L + y_1, \quad (5)$$

причем $|y_1| \ne |y_2|, |y_1|, |y_2| < L$.

Положим, что на основе обработки потока сигналов $W^{\nu}W^{\nu}...W^{\nu}$ принимается как истинный либо сигнал

$$W^{\vartheta}_{x_L} = W^{\vartheta}_{L-\delta} W^{\vartheta}_L W^{\vartheta}_{x_1 - L - \delta}, \quad (6)$$

либо

$$W^{\vartheta}_{x_1} = W^{\vartheta}_{L-\delta} W^{\vartheta}_{X_1+\delta} \quad (7)$$

при $\Delta x \ge L$, либо сигнал

$$W^{\vartheta}_{x_2} = W^{\vartheta}_{L-X_2}, \quad (8)$$

либо

$$W^{\vartheta}_{x_{21}} = W^{\vartheta}_{\delta} W^{\vartheta}_{L-x_2-\delta} \quad (9)$$

при $\Delta x < L$, где индексы x_1 и x_2, δ, L, $x_1 + \delta - L$, $L - \delta$, $x_1 + \delta$, $L - x_2 - \delta$ указывают число символов усеченного сигнала W^{ϑ} (первых или последних соответственно расположению его символов $W^{\vartheta}_{X_1}$ или $W^{\vartheta}_{X_2}$. Тогда размытость сигналов, заданных (6 – 9) будем представлять совокупностью систем нелинейных параметрических неравенств:

$$R_{a_1}(k) \le \sum_{i=\delta}^{L-K} W_i^{\vartheta} \left(W^{\vartheta}_{i+k}\right)^* + \sum_{i=L-k+1}^{L} W_i^{\vartheta} \left(W^{\vartheta}_{i-L+K}\right)^* + \sum_{i=1}^{L-K} W_i^{\vartheta} \left(W^{\vartheta}_{i+k}\right) + \sum_{i=L-k+1}^{L} W_i^{\vartheta} \left(W^{\vartheta}_{i-L+K}\right)^* + \sum_{i=1}^{x_1 - L + \vartheta} W_i^{\vartheta} \left(W^{\vartheta}_{i+k}\right)^* \le R^{'}_{a_2}(k);$$

$$k = \overline{0, L + x_2}, \quad \text{а)}$$

$$R_{a_1}(k) \le \sum_{i=\delta}^{L-K} W_i^{\vartheta} \left(W^{\vartheta}_{i+k}\right)^* + \sum_{i=L-k+1}^{L} W_i^{\vartheta} \left(W^{\vartheta}_{i-L+K}\right)^* + \sum_{i=1}^{L-K} W_i^{\vartheta} \left(W^{\vartheta}_{i+k}\right) + \sum_{i=L-k+1}^{L} W_i^{\vartheta} \left(W^{\vartheta}_{i-L+K}\right)^* \le R^{'}_{a_2}(k);$$

$$k = \overline{0, L + x_1}, \quad \text{б)}$$

$$R_{a_2}(k) \le \sum_{i=1}^{L-x_1} W_i^{\vartheta} \left(W^{\vartheta}_{i-k}\right)^* \le R^{'}_{a_2}(k), k = \overline{0, L - x_2}, \quad \text{в)}$$

$$R_{a_1}(k) \le \sum_{i=L-\delta}^{L-K} W_i^{\vartheta} \left(W^{\vartheta}_{i+k}\right)^* + \sum_{i=L-k+1}^{L} W_i^{\vartheta} \left(W^{\vartheta}_{i-L+K}\right)^* + \sum_{i=1}^{L-x_2+\vartheta} W_i^{\vartheta} \left(W^{\vartheta}_{i+k}\right)^* \le R^{'}_{a_2}(k);$$

$$k = \overline{0, L - x_2}, \quad \text{г) (10)}$$

где $R^{'}_{a_1}(k)$ и $R^{'}_{a_2}(k)$ – различные реализации ПФАК, задаваемые при синтезе сигналов.

В случае размытости по ПФВК и СФВК в интервале Δx, определяемого как:

$$L - x_2 \le \Delta x \le L + x_1,$$

размытость может быть задана совокупностью систем нелинейных неравенств

$$R^{'}_{b_1}(k) \le \sum_{i=\delta}^{L-K} W_i^q \left(W^{\vartheta_1}_{i+k}\right)^* + \sum_{i=L-k+1}^{L} W_i^q \left(W^{\vartheta_2}_{i-L+K}\right)^* + \sum_{L=1}^{L-K} W_i^p \times \left(W^{\vartheta_2}_{L+k}\right)^* + \sum_{i=L-K+1}^{L} W_i^p \left(W^{\vartheta_3}_{i-L+K}\right)^* + \sum_{i=1}^{L-K} W_i^r \times \left(W^{\vartheta_3}_{i+K}\right)^* \le R^{'}_{b_2}(k);$$

$$k = \overline{0, L + x}, \quad \text{а)}$$

$$R^{'}_{b_1}(k) \le \sum_{i=\delta}^{L-K} W_i^q \left(W^{\vartheta_1}_{i+k}\right)^* + \sum_{i=L-k+1}^{L} W_i^q \left(W^{\vartheta_2}_{i-L+K}\right)^* + \sum_{L=1}^{L-K} W_i^p \times \left(W^{\vartheta_2}_{L+k}\right)^* + \sum_{i=L-K+1}^{L} W_i^p \left(W^{\vartheta_3}_{i-L+K}\right)^* \le R^{'}_{b_2}(k);$$

$$k = \overline{0, L + x}, \quad \text{б)}$$

$$R^{'}_{b_2}(k) \le \sum_{i=L-\delta}^{L-K} W_i^q * (W_i^{\vartheta_1} + k)^* \le R_{b_2}(k), k = \overline{0, L - x_2}, \quad \text{в)}$$

$$R^{'}_{b_1}(k) \le \sum_{i=L-\delta}^{L-K} W_i^q \left(W^{\vartheta_2}_{i+k}\right)^* + \sum_{i=L-k+1}^{L} W_i^q \left(W^{\delta_2}_{i-L+K}\right)^* +$$

$$+\sum_{i=1}^{L-x_2+\delta} W_i^p\left(W_{i+k}^{\vartheta_2}\right)^* \le R_{b_2}^{'}(k);$$

$$k=\overline{0,L-x_2}, \qquad \text{г) (11)}$$

Таким образом, условие которое должно выполняться для синтезируемой системы сигналов W_m^q, может быть сформулировано следующим образом: словарь $\{W_m^q\}$ удовлетворяет совокупности систем нелинейных параметрических неравенств (10) – (11), т.е. словарь $\{W_m^q\}$ обладает в интервалах Δx и Δy разностью по длительности и цикловой частоте.

4. В каждом из М словарей существуют сигналы $W_{m_1}^{q_1}$ и $W_{m2}^{q_2}$, авто - и взаимная свертка которых удовлетворят совокупности неравенств вида (2) и (3);

5. Закон формирования каждого из сигналов W_m^q может быть определен при перехвате не менее L сигналов, то есть по критерию (1) W_m^q обладает структурной скрытностью.

6. Апериодическая нормированная автосвертка W_m^q удовлетворяет системе нелинейных неравенств

$$r_{a_1}^q(l) \le \sum_{i=1}^{L-m} W_i^q\left(W_{i+1}^q\right)^* \le r_{a_2}^q(l); \qquad (12)$$

$$l=\overline{1,L},\ \ m=\overline{1,L},$$

где $r_{a_1}^q(l)$ и $r_{a_2}^q(l)$ – заданные реализации АФАК.

7. Апериодическая взаимная свертка удовлетворяет двум системам нелинейных параметрических неравенств

$$r_{b_{1,1}}^{qp}(l) \le \frac{1}{L-m}\sum_{i=0}^{L-m} W_i^q\left(W_{i+1}^q\right)^* \le r_{b_{1,2}}^{qp}(l);$$

$$l=\overline{1,L},\ \ m=\overline{1,L},$$

$$r_{b_{2,1}}^{qp}(l) \le \frac{1}{L-m}\sum_{i=0}^{L-m} W_i^p\left(W_{i+1}^q\right)^* \le r_{b_{2,2}}^{pq}(l);$$

$$l=\overline{1,L},\ \ m=\overline{1,L}, \qquad (13)$$

8. Целевая функция

$$Int(E)=\sum_{j=1}^{n} C_j S_j \qquad (14)$$

принадлежит интервалу (A,B), где S_j – значения реализаций функций системы передачи информации, описывающих законы распределения величин апериодических и периодических функций корреляции, определяющих структурную скрытность сигналов, алгоритмы построения ДС и др., а C_j – соответствующие им штрафы.

Сформулируем задачу синтеза системы сигналов, с учетом основного отличия от системы сигналов, рассмотренной ранее, – длительности некоторых (всех) векторов (сигналов) W^q в каждом из словарей отличаются относительно средней длительности L_{cp} на величину $\pm\Delta L$. Назовем такую систему системой нелинейных сигналов (НС).

Пусть источник дискретных сигналов (ДС) Q_m с максимальной энтропией $H(Q_m=\log p^{L_{cp}})$ выдает L_j -значные над полем $GF(P)$ последовательности такие, что для некоторых или для всех сигналов выполняется условие: $L_i \ne L_j$, $i,j=\overline{1,N}$, $i \ne j$), тогда под задачей синтеза НС сигналов будем понимать задачу построения словарей (подмножеств) векторов $\{W_m^q\}$, $q=\overline{1,H}, m=\overline{1,M}$, вся совокупность которых образует систему НС сигналов, удовлетворяющих следующим условиям.

1. Автосвертка или периодическая функция автокорреляции (ПФАК) каждого из W_m^q ДС удовлетворяет системе нелинейных параметрических неравенств вида (2).

2. Истинными являются условия (12 - 14).

3. Взаимная свертка или стыковые функции автокорреляции (СФВК) $W^q(W^p)$ ДС со стыковыми словарями $W^{pp}(W^{qq}), W^{qp}, W^{pq}$, при условии, что $L_q < L_p$, удовлетворяет совокупности систем нелинейных параметрических неравенств

$$\begin{cases} R_{b_{1,1}}(0) \le W_1^q W_1^p + W_2^q W_2^p + \ldots + W_\delta^q W_\delta^p + \ldots + \\ R_{b_{1,1}}(1) \le W_1^q W_2^p + W_2^q W_3^p + \ldots + W_\delta^q W_{\delta+1}^p + \ldots + \\ R_{b_{1,1}}(2) \le W_1^q W_3^p + W_2^q W_4^p + \ldots + W_\delta^q W_{\delta+2}^p + \ldots + \\ R_{b_{1,1}}(\xi) \le W_1^q W_{\xi+1}^p + W_2^q W_{\xi+2}^p + \ldots + W_\delta^q W_{\xi+\delta}^p + \ldots + \\ R_{b_{1,1}}(Lp) \le W_1^q W_{Lp}^p + W_2^q W_1^p + \ldots + W_\delta^q W_{\xi-1}^p + \ldots + \end{cases}$$

$$\begin{aligned} &+W_{Lq}^q W_{Lq}^p \le R_{b_{2,1}}(0), a) \\ &+W_{Lq}^q W_{Lq+1}^p \le R_{b_{2,1}}(1), a') \\ &+W_{Lq}^q W_{Lq+2}^p \le R_{b_{2,1}}(2), \hat{a}) \\ &+W_{Lq}^q W_{\xi}^p \le R_{b_{2,1}}(\xi), \tilde{a}) \\ &+W_{Lq}^q W_{Lp-1}^p \le R_{b_{2,1}}(Lp), \ddot{a}) \end{aligned} \qquad (15)$$

$$\begin{cases} R_{b_{2,1}}(1) \le W_1^q W_2^q + W_2^q W_3^q + \ldots + W_\delta^q W_{\delta+1}^p + \ldots + \\ R_{b_{1,2}}(2) \le W_1^q W_3^q + W_2^q W_4^q + \ldots + W_\delta^q W_{\delta+2}^p + \ldots + \\ R_{b_{1,2}}(3) \le W_1^q W_4^q + W_2^q W_5^p + \ldots + W_\delta^q W_{\delta+3}^q + \ldots + \\ R_{b_{1,2}}(\xi) \le W_1^q W_{\xi+1}^p + W_2^q W_{\xi+2}^p + \ldots + W_\delta^q W_{\xi+\delta}^p + \ldots + \\ \ldots. \\ R_{b_{1,2}}(Lq-1) \le W_1^q W_{Lp}^p + W_2^q W_1^p + \ldots + W_\delta^q W_{\delta-1}^p + \ldots + \end{cases}$$

$$\begin{aligned} &+W_{Lq}^q W_1^p \le R_{b_{2,2}}(1), a) \\ &+W_{Lq}^q W_2^p \le R_{b_{2,2}}(2), б) \\ &+W_{Lq}^q W_3^p \le R_{b_{2,2}}(3), в) \\ &+W_{Lq}^q W_{\xi}^p \le R_{b_{2,2}}(\xi), г) \\ &\ldots. \\ &+W_{Lq}^q W_{Lq-1}^p \le R_{b_{2,2}}(Lq-1), д) \end{aligned} \qquad (16)$$

$$R_{b_{1,3}}(0) \le W_1^q W_1^p + W_2^q W_2^p + \ldots +$$

$$+W_\delta^q W_\delta^p + \ldots + W_{Lq}^q W_{Lq}^p \le R_{b_{2,3}}(0),$$

$$R_{b_{1,3}}(1) \le W_1^q W_2^p + W_2^q W_3^p + \ldots + W_\delta^q W_{\delta+1}^p +$$
$$+ \ldots + W_{Lq}^q W_{Lq+1}^p \le R_{b_{2,3}}(1),$$
$$R_{b_{1,3}}(Lp - Lq + 1) \le W_1^q W_{Lp-Lq+2}^p +$$
$$+ W_2^q W_{Lp-Lq+3}^p + \ldots + W_\delta^q W_{Lp-Lq+\delta-1}^p + \ldots +$$
$$+ W_{Lq}^q W_1^q \le R_{b_{2,3}}(Lp - Lq + 1), \qquad (в)$$
$$R_{b_{1,3}}(Lp) \le W_1^q W_{Lp}^p + W_2^q W_1^p + \ldots + W_p^q W_{\delta-1}^p + \ldots +$$
$$+ W_{Lq}^q W_{Lq-1}^q \le R_{b_{2,3}}(Lp), \qquad (г),$$

для всевозможных сочетаний q и p, причем $q, p = \overline{1, N}$, $q \ne p$, где

$$R_{b_{1,1}}(l), R_{b_{1,2}}(l), R_{b_{1,3}}(l), R_{b_{2,1}}(l), R_{b_{2,2}}(l), \text{ и } R_{b_{2,3}}(l),$$

– значения реализаций периодической функции взаимной корреляции (ПФВК) и СФВК.

4. Выполняются условия (15) и (16) для апериодических авто- и взаимных сверток для всех W^q, $q = \overline{1, N}$ любого сочетания ДС W^q и W^p, $q, p = \overline{1, N}, q \ne p$, целевая функция (14) принадлежит интервалу (A, B).

Подчеркнем, что приведенная постановка задач синтеза НС является более общей, чем постановка задач синтеза РС (вставить обозначение РС в 1 ю часть статьи) сигналов. Подчеркнем также, что как сама постановка задачи синтеза РС (НС) систем сигналов, так и развиваемый подход являются новыми. Поэтому получение даже частных решений позволяет в дальнейшем продвинуться в направлении решения задач синтеза ДС с заданными корреляционными ансамблевыми и структурными свойствами.

В [2] показано, что улучшение ансамблевых, структурных и корреляционных свойств ДС при несущественном усложнении алгоритмов и устройств их формирования, может быть достигнуто на основе применения так называемых составных систем (СС) сигналов. При этом составными будем называть системы сигналов по двум причинам. Во-первых, закон формирования сложных элементов в составном сигнале может изменяться, а во-вторых, сложные элементы, образующие СС, обладают идентичными (близкими) авто- и взаимными корреляционными свойствами, поэтому их взаимное комбинирование не приводит к ухудшению корреляционных свойств и в тоже время позволяет улучшить ансамблевые свойства, повысить структурную скрытность и реализовать режим бегущий код без особого усложнения устройств формирования и обработки.

Сформулируем задачу синтеза СС сигналов. Пусть источник ДС формирует M РС или НС систем сигналов каждая объема N_j, для которых выполняются условия (2)–(3), (12)–(13), и (15)–(16), тогда под задачей построения системы СС сигналов будем понимать процедуру комбинирования сложных элементов, являющихся РС или НС сигналами, при которых каждый из СС сигналов содержит m сложных элементов и выполняются условия:

1. Целевая функция вида (14) при заданной (заранее выбранной) матрице штрафов принадлежат интервалу (A_c, B_c);

2. Авто – и взаимные свертки СС $W_i^{q_c}$ (ПФАК, ПФВК) с точки зрения максимально допустимых боковых выбросов и дисперсии σ_R не зависят от типа конфигурации образования СС сигнала.

3. Закон формирования всех m сложных сигналов W_i^q СС сигнала изменяется в каждом из сложных элементов.

В такой постановке задача построения СКС систем сигналов сводится к поэтапному решению задач синтеза РС или НС систем сигналов.

Проведенный анализ показал, что решение задач синтеза РС, НС и СС систем сигналов прежде всего связано с исследованием алгебраической структуры систем нелинейных параметрических неравенств (2) – (3), (12) – (14), разработкой подходов и теоретических основ их решения.

Рассмотрим вначале теоретические основы синтеза двух РС сигналов x^q и x^p, не накладывая ограничений размытости вида (10) и (11), а затем сделаем ряд обобщений на случай синтеза N дискретных сигналов, обладающих в том числе и размытыми свойствами. При этом потребуем, чтобы по критерию (1) РС сигналы обладали идеальными структурными свойствами, т.е. такой структурной скрытностью, что при перехвате и поэлементной обработке любого числа l символов РС сигналов нельзя однозначно предсказать оставшимся $L-l$ символов. Это может быть выполнено, если символы в РС сигналах независимы и появляются с равной вероятностью.

С учетом систем вида (2) – (3), для случая синтеза двух дискретных сигналов, совокупность систем нелинейных неравенств имеет вид:

$$\xi^1 a_1(l) \le \sum_{i=1}^{L} x_i^q * (x_{i+l}^q)^* \le \xi^1 a_2(l), l = \overline{0, L-1} \qquad а)$$
$$\xi^2 a_1(l) \le \sum_{i=1}^{L-K} x_i^p * (x_{i+l}^p)^* \le \xi^2 a_2(l), l = \overline{0, L-1} \qquad б)$$
$$\xi^1 b_1(l) \le \sum_{i=0}^{L-K} x_i^q * (x_{i+l}^p)^* + \sum_{i=L-K+1}^{L-K} x_i^q * (x_{i-L+K}^p)^* \le$$
$$\le \xi^1 b_2(l),\ l = \overline{0, L-1} \qquad в)$$
$$\xi^2 b_1(l) \le \sum_{i=0}^{L-K} x_i^p * (x_{i+l}^p)^* + \sum_{i=L-K+1}^{L-1} x_i^p * (x_{i-L+K}^p)^* \le$$
$$\le \xi^2 b_2(l),\ l = \overline{0, L-1} \qquad г)$$
$$\xi^3 b_1(l) \le \sum_{i=0}^{L-K} x_i^q * (x_{i+l}^q)^* + \sum_{i=L-K+1}^{L-1} x_i^q * (x_{i-L+K}^p)^* \le$$
$$\le \xi^3 b_2(l),\ l = \overline{0, L-1} \qquad д) \quad (17)$$
$$\xi^4 b_1(l) \le \sum_{i=0}^{L-K} x_i^q * (x_{i+l}^p)^* + \sum_{i=L-K+1}^{L-1} x_i^p * (x_{i-L+K}^p)^* \le$$

$$\leq \xi^4 b_2(l), \quad l = \overline{0, L-1} \qquad \text{е)}$$

$$\xi^5 b_1(l) \leq \sum_{i=0}^{L-K} x_i^p * (x_{i+l}^p)^* + \sum_{i=L-K+1}^{L-1} x_i^p * (x_{i-L+K}^q)^* \leq$$

$$\leq \xi^5 b_2(l), \; l = \overline{0, L-1} \qquad \text{ж)}$$

$$\xi^6 b_1(l) \leq \sum_{i=0}^{L-K} x_i^p * (x_{i+p}^q)^* + \sum_{i=L-K+1}^{L-1} x_i^p * (x_{i-L+K}^p)^* \leq$$

$$\leq \xi^6 b_2(l), \; l = \overline{0, L-1} \qquad \text{з)}$$

В приведенной постановке задача синтеза словарей одного класса сигналов является наиболее обобщенной, так как в ней ставится задача синтеза систем сигналов с заданными корреляционными свойствами, законами формирования, структурными и ансамблевыми свойствами. В частности, необходимо, на наш взгляд, провести исследования алгебраической структуры систем нелинейных параметрических неравенств вида (17).

Литература

[1] *Ipatov, Valery P.* Spread Spectrum and CDMA. Principles and Applications [Текст] / Valery P. Ipatov. University of Turku, Finland and St. Petersburg Electrotechnical University 'LETI', Russia. – John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex PO19 8SQ, England, 2005. – 385 p.

[2] *Горбенко И.Д.* Алгоритм построения многопозиционных характеристических дискретных сигналов / Радиотехника. Вып. № 101, 1997. С. 3–10.

[3] *Свердлик М.Б.* Оптимальные дискретные сигналы. – М., 1975. – 200 с.

Поступила в редколлегию 3.04.2012

Горбенко Иван Дмитриевич, фото и сведения об авторе см. на с. 190.

Замула Александр Андреевич, фото и сведения об авторе см. на с. 193.

УДК 621.391.1

Синтез систем сигналів із заданими кореляційними властивостями, законами формування, структурними і ансамблевими властивостями / І.Д. Горбенко, О.А. Замула // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 293–298.

Розглядається задача синтезу дискретних сигналів із заданими кореляційними, структурними і ансамблевими властивостями.

Ключові слова: ансамбль складних сигналів, перешкодостійкість, мінімаксний критерій, синтез системи сигналів.

Бібліогр.: 03 найм.

UDC 621.391.1

Synthesis of systems of signals with set cross-correlation properties, forming laws, structural and band properties / I.D. Gorbenko, A.A. Zamula // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 293–298.

The problem of synthesis of discrete signals with given correlation, structural and ensemble properties is considered.

Keywords: ensemble of compound signals, interference immunity, minimax criterion, synthesis of a system of signals.

Ref.: 03 items.

УДК 681.3.06

ВЛАСТИВОСТІ ДІЯЛЬНОСТІ ІЗ ЗАБЕЗПЕЧЕННЯ ЗАХИСТУ ІНФОРМАЦІЇ ЯК СИСТЕМНОЇ КАТЕГОРІЇ

О.В. ПОТІЙ, Д.Ю. ПИЛИПЕНКО, Д.І. ГЛАДКИЙ

В роботі досліджуються властивості діяльнісного аспекту захисту інформації як системної категорії. Сформована множина властивостей дозволяє різнобічно оцінювати діяльність із захисту інформації. Наведена множина властивостей також становить певну основу для подальшого дослідження та розробки аналітичних виразів та критеріїв оцінки організаційної системи захисту інформації.

Ключові слова: системодіяльнісний підхід, діяльність із захисту інформації, оцінювання.

ВСТУП

У більшості випадків захист інформації як об'єкт досліджень не враховує аспект людської діяльності, або ж йому приділяється не достатньо велика увага. Проте, діяльнісний аспект відіграє значну роль в інформаційній безпеці, і сьогодні можна із впевненістю казати, що серед фахівців формується певний інтерес до розглядання проблем інформаційної безпеки у рамках системодіяльнісного підходу.

Розглядаючи захист інформації як діяльність, слід зазначити, що ця системна категорія характеризується певними властивостями та характеристиками. Властивості – це об'єктивні особливості діяльності, які проявляються під час її здійснення. Захист інформації в свою чергу може бути охарактеризований з точки зору ефективності, цілеспрямованості, безперервності, організованості, керованості, узгодженості тощо. Виявлення, розкриття змісту, систематизація та опис цих та інших властивостей діяльності, визначення показників та критеріїв їх оцінювання, формування відповідних теоретичних, науково-методичних основ та розробка інструментарію оцінювання властивостей та характеристик є окремою важливою та актуальною дослідницькою задачею.

1. ОНТОЛОГІЯ ВЛАСТИВОСТЕЙ ЗАХИСТУ ІНФОРМЦАЇ

Як зазначено на рис.1, діяльнісний аспект захисту інформації здійснюється у рамках організаційної системи захисту інформації. Діяльність має певні властивості, а саме: ефективність, зрілість, адекватність, прийнятність, гнучкість, здійсненність та простоту в адміністративному забезпеченні. Ефективність є досить широкою властивістю, що дозволяє досліджувати її на більш глибокому рівні. Ефективність захисту інформації можна розглядати з точки зору результативності, економічності та оперативності.

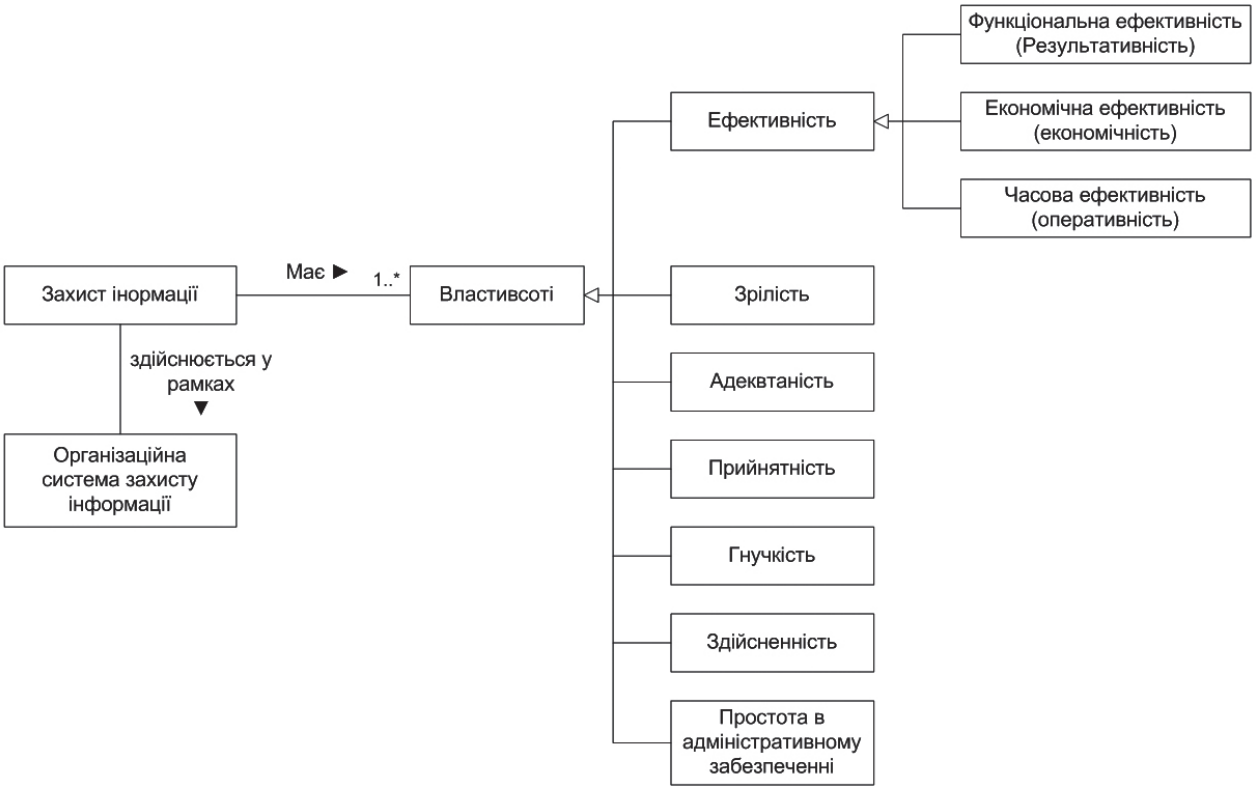


Рис. 1. Проблемно-орієнтовна онтологія властивостей захисту інформації як діяльності

2. ЕФЕКТИВНІСТЬ ЗАХИСТУ ІНФОРМАЦІЇ

Одним з найважливіших питань у рамках системодіяльнісної методології є визначення поняття та змісту ефективності захисту інформації. Підвищення ефективності захисту інформації саме як діяльності, тобто у рамках організаційної системи захисту інформації, підвищення ефективності організаційних форм та методів захисту інформації є актуальною задачею як у практиці, так і у теорії захисту інформації. У визначенні ефективності захисту інформації, як діяльності виходитимемо з таких поглядів.

Будь-яка діяльність здійснюється у рамках деякої організаційної системи S_0. По відношенню до цієї системи мета діяльності X_0 є основним системоутворюючим чинником як спосіб інтеграції різних видів діяльності (процесів, дій) у єдину систему. Мета є ідеальним уявленням у свідомості суб'єкта діяльності бажаного результату. Вона визначає способи та форми організації діяльності, характер та системну впорядкованість, а також засоби досягнення мети суб'єктом [1].

Взагалі будь-яка мета у формалізованому вигляді може бути виражена деяким набором бажаних параметрів Y_{mp}. Реальний результат Y діяльності у загальному випадку може не співпасти з тим, що вимагається. У відповідності до усталених поглядів, під ефективністю розуміють співвідношення бажаного результату Y_{mp} до реально досягнутого результату Y. Результат діяльності Y ставлять у залежність від трьох основних результуючі факторів – корисного ефекту Q, затраченого ресурсу C та часу T. Ці фактори залежать від конкретної обраної стратегії u, тобто [1]:

$$Y(u) = Y\big(Q(u), C(u), T(u)\big).$$

Головною метою захисту інформації є досягнення певного рівня захищеності інформації або безпеки інформації. Тому результатом діяльності із захисту інформації, корисним ефектом є певний досягнутий рівень безпеки інформації L_{SI}. На досягнення певного рівня безпеки інформації, суб'єкт захисту витрачає матеріальні, фінансові, людські та інші ресурси C та час T. Рівень безпеки, витрати ресурсів та часу безпосередньо залежать від конкретної обраної суб'єктом захисту стратегії захисту U_{SP}. Таким чином, результат захисту інформації може бути поданий як

$$Y(U_{SP}) = Y(L_{SI}(U_{SP}), C(U_{SP}), T(U_{SP})).$$

Під ефективністю захисту інформації розумітимемо співвідношення бажаного результату захисту інформації (у термінах рівня безпеки інформації, витрат ресурсів та часу) та реально досягнутого результату.

Спираючись на підходи щодо класифікації ефективності у різних сферах діяльності [1, 2, 3, 4] у подальшому пропонується розрізнювати функціональну ефективність, економічну ефективність та часову ефективність захисту інформації.

Функціональна ефективність або *результативність* захисту інформації є ступінь відповідності реального рівня безпеки інформації L_{SI} тому, що очікувався або вимагався L^*_{SI}, тобто

$$Э_f(U_{SP}) = f(L_{SI}(U_{SP}), L^*_{SI}).$$

Під *економічною ефективністю* або *економічністю* захисту інформації розумітимемо співвідношення досягнутого рівня безпеки інформації L_{SI} та затраченого ресурсу C, тобто

$$Э_\rho(U_{SP}) = \rho(L_{SI}(U_{SP}), C(U_{SP})).$$

Під *часовою ефективністю* або *оперативністю* розумітимемо співвідношення досягнутого рівня безпеки інформації L_{SI} до часових витрат T, що понесені на досягнення цього рівня, тобто

$$Э_\tau(U_{SP}) = \tau(L_{SI}(U_{SP}), T(U_{SP})).$$

Таким чином, ефективність захисту інформації є узагальнена визначальна функціональна властивість системи захисту інформації, у рамках якої здійснюється ця діяльність, яка з гносеологічної точки зору розкривається через категорію мети (тобто рівня безпеки) та об'єктивно виражається ступенем досягнення мети захисту інформації X_0 з урахуванням витрат ресурсів C та часу T:

$$Э_{SP}(U_{SP}) = F(Э_f(U_{SP}), Э_\rho(U_{SP}), Э_\tau(U_{SP})).$$

Діяльність із захисту інформації містить адміністративну, процедурну та технічну складові. Виходячи з цього, ми можемо розглядати ефективність управління захистом інформації Y_{man} (ефективність системи управління захистом інформації), ефективність організації захисту інформації Y_{op} (ефективність організаційної системи захисту інформації) та ефективність технічних систем у захисті інформації Y_{tech}. Для кожної з цих видів ефективності цілком логічно розглядати функціональну, економічну та часову ефективності (рис. 2).

У галузі захисту інформації досить глибокий розвиток знайшли питання ефективності технічних, криптографічних, фізичних систем захисту, комплексів, механізмів та способів захисту інформації, що відповідає сучасному розвитку методологічних підходів до захисту інформації. Останнім часом у роботах фахівців, у міжнародних та національних стандартах піднімаються питання управління захистом інформації, починають формулюватися задачі в сфері ефективності управління захистом інформації. Але питання ефективності діяльності із захисту інформації, питання раціональної організації цієї діяльності поки що не знайшли відповідного відображення у сучасних роботах з теорії захисту інформації.

3. ЗРІЛІСТЬ ЗАХИСТУ ІНФОРМАЦІЇ

Зрілість захисту інформації об'єднує множину властивостей які охоплюють аспекти розвитку, еволюції захисту інформації, як діяльності, характеризують функціональні можливості організаційної системи захисту інформації.

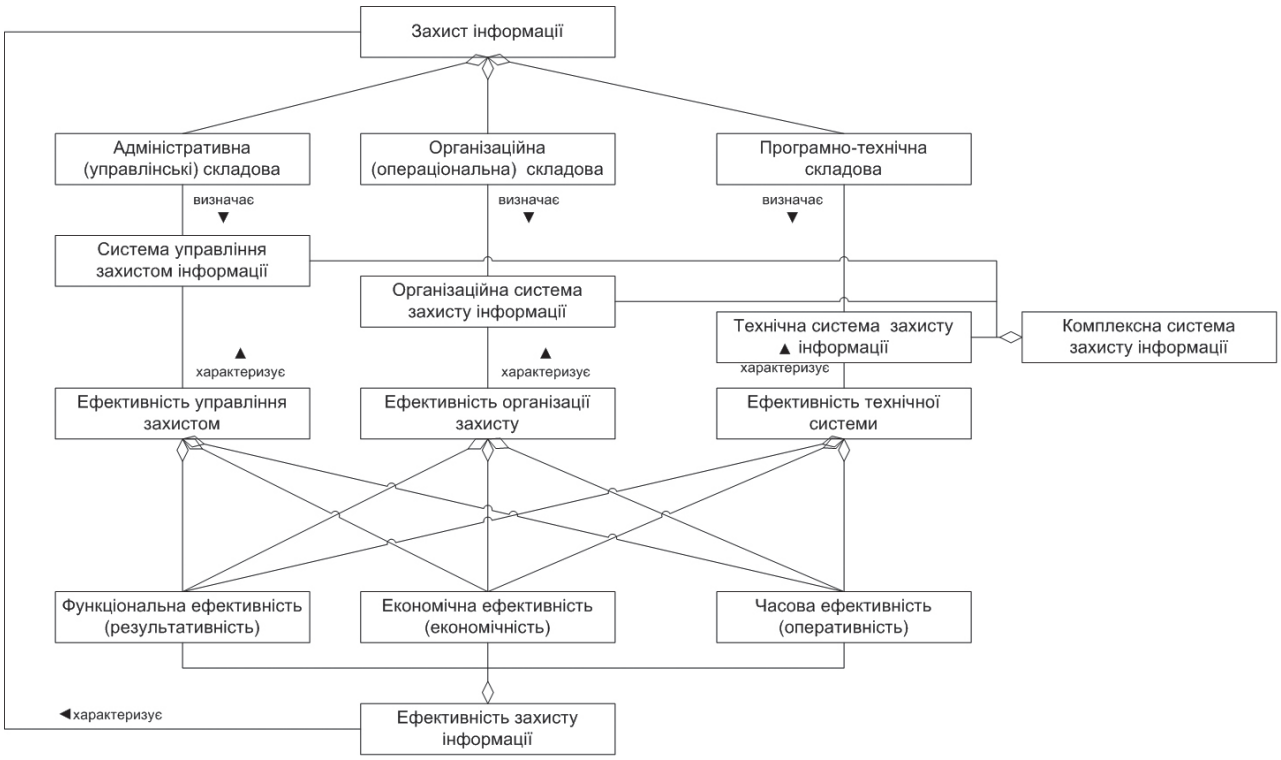


Рис. 2. Проблемно-орієнтовна онтологія ефективності захисту інформації

4. АДЕКВАТНІСТЬ ЗАХИСТУ ІНФОРМАЦІЇ

Під *адекватністю* захисту інформації розумітимемо співвідношення рівня функціональної ефективності захисту інформації та рівня небезпеки (ризику) інформаційним активам та іншим ресурсам, що пов'язані з обробкою інформації:

$$A = F_A(\text{Э}_f, R).$$

Адекватність визначає, що даний рівень безпеки інформації L_{SI} дійсно відповідає ситуації, яка склалася відносно існуючих загроз та рівня ризиків безпеці і задовольняє потреби власника інформаційних ресурсів відносно забезпечення безпеки інформації. Захист інформації є адекватним, якщо правила безпеки, рівень зрілості процесів захисту інформації, рівень суворості заходів захисту, рівень гарантій (довіри) захисту інформації, механізми захисту інформації та рівень їх реалізації є адекватними загрозам та ризикам безпеки для певного об'єкта захисту.

5. ПРИЙНЯТНІСТЬ ЗАХИСТУ ІНФОРМАЦІЇ

Прийнятність пов'язана з визначенням несуперечливості задач і цілей захисту інформації місії та загальним цілям організації (об'єкта захисту).

Нехай цілі та задачі захисту інформації будуть подані у вигляді вектора $G_{SI} = \left\{g_1^{SI}, g_2^{SI} \ldots g_n^{SI}\right\}$, а загальні цілі організації у вигляді вектора $G_S = \left\{g_1^S, g_2^S \ldots g_m^S\right\}$. Тоді під прийнятністю захисту інформації розумітимемо ступінь відповідності цілей та задач захисту інформації G_{SI} цілям (місії) G_S організації, у якій здійснюється діяльність із захисту інформації та яка характеризується коефіцієнтом прийнятності виду:

$$K_{accept} = F_{accept}\{G_{SI}, G_S\},$$

де F_{accept} – оператор визначення узгодженості (несуперечливості) цілей захисту та загальних цілей. Таку функцію можна використовувати, наприклад, як неформальні методики цільового аналізу, онтологічного аналізу тощо. Коефіцієнт K_{accept} може бути виражений, наприклад, як лінгвістична змінна рівня узгодженості.

Оцінка прийнятності уточнює питання, чи необхідні цілі і задачі захисту у конкретній організаційній системі, чи не має протиріччя між загальними задачами, що стоять перед нею? Для оцінки захисту інформації за даним критерієм необхідно врахувати усі показники одночасно, виразити відношення між їхніми кількісними формами. Використання для проектування захисту інформації стандартів безпеки або залучення до розв'язання задач захисту експертів – це один з основних доводів прийнятності захисту інформації. Таким чином, захист інформації є прийнятним, якщо цілі і задачі захисту не мають протиріч між місією та основними задачами організації (об'єктів захисту), узгоджені з цілями функціонування організації.

6. ГНУЧКІСТЬ ЗАХИСТУ ІНФОРМАЦІЇ

Гнучкість системи це властивість системи, що полягає у можливості її удосконалення, розширення та надання їй нових якостей [5]. Гнучкість передбачає легку, тобто без великих зусиль та витрат, та достатньо швидку зміну стану системи. Гнучкість, як властивість комплексної системи захисту інформації (КСЗІ), була розглянута у роботі [6]. Інтенсивна зміна середовища безпеки (оточення), умов здійснення діяльності із захисту

інформації призводить до того, що організаційна система захисту інформації (ОСЗІ) має пристосовуватися (адаптуватися) до розв'язання нових задач захисту, що володіють певним ступенем різноманіття. Гнучкість, як економічна категорія відображає здатність ОСЗІ змінювати свої цілі без суттєвих витрат [6]. Спираючись на результати, що отримані у [6] пропонується ввести показник гнучкості ЗІ у вигляді:

$$K_{flex}(t_н,t_к)=\frac{I_{obj}(t_н,t_к)}{I\varsigma(t_н,t_к)},$$

де $K_{flex}(t_н,t_к)$ – гнучкість захисту інформації у період $(t_н,t_к)$; $I_{obj}(t_н,t_к)$ – індекс, що характеризує зміну ступеню різноманітності задач захисту, що розв'язується КСЗІ у період $(t_н,t_к)$; $I\varsigma(t_н,t_к)$ – індекс, що характеризує зміну зведених економічних витрат, що пов'язані із захистом інформації у період $(t_н,t_к)$. Чим більше значення $K_{flex}(t_н,t_к)$, тим захист інформації є більш гнучким.

Гнучкість (реагувальність) пов'язано з оцінкою рівня можливостей (здатності) задовольнити потреби власника інформаційних ресурсів у відношенні безпеки інформації в умовах обстановки, що змінюються, під час здійснення діяльності із захисту інформації та реалізації процесів захисту. Показник гнучкості залежить від індексу зміни ступеня різноманіття задач захисту $I_{obj}(t_н,t_к)$. Серед чинників, що впливають на цей індекс можна виділити такі:

– ступінь варіативності цілей функціонування надсистеми (наприклад бізнес-цілей, цілей автоматизованого управління військами тощо), до якої входить КСЗІ як інтегрована частина.

– варіативність оточення безпеки. Мінливість оточення складається з можливості зміни ресурсів зловмисника, що у свою чергу призводить до зміни кількісного та якісного складу загроз безпеки, зміни ресурсів, що захищаються, сукупність яких визначається з ситуації, що склалася.

Таким чином, захист інформації має бути здатним адекватно реагувати на зміни умов функціонування організації, на зміни цілей і задач функціонування, інтересів і потреб власника інформаційних ресурсів. Захист інформації називається гнучким, якщо він здатний задовольнити потреби у безпеці інформації у будь-яких умовах функціонування організації.

7. ЗДІЙСНЕННІСТЬ ЗАХИСТУ ІНФОРМАЦІЇ

Здійсненність пов'язана з визначенням здійсненності конкретних процесів захисту інформації P у конкретних умовах $X=\{x_1,x_2,...,x_n\}$ при заданих обмеженнях $Y=\{y_1,y_2,...,y_n\}$ у конкретній організації S_0. Необхідно враховувати умови здійсненності захисту інформації і впровадження процесів захисту інформації на різних рівнях забезпечення безпеки інформації. Безпека інформації забезпечується на чотирьох рівнях – законодавчому (правовому), адміністративному, процедурному і програмно-технічному.

На правовому рівні необхідно оцінювати процеси захисту інформації з точки зору їх легітимності. Чи можуть конкретні процеси захисту інформації бути реалізовані на даному об'єкті з точки зору правового поля держави в галузі захисту інформації? Чи має право керівництво приймати такого роду рішення? Чи відповідають положення політики безпеки нормам законів та інших нормативних актів у галузі захисту інформації.

Здійсненність процесів захисту інформації на адміністративному рівні залежить від ступеня розуміння керівництвом цілей безпеки і задач захисту, усвідомлення реальності загроз безпеці, реалізація яких може нанести збиток, рівня сформованості потреб у розв'язанні таких задач захисту.

На процедурному рівні здійсненність процесів захисту інформації залежить від ступеня технологічної і організаційної готовності об'єкта інформатизації до їх впровадження. Тут важливе місце набуває готовність персоналу виконувати конкретні роботи та завдання. Така готовність залежить від багатьох чинників: розуміння персоналом необхідності виконання вимог безпеки, рівень усвідомлення і дисциплінованості персоналу, рівень його професійної підготовленості.

На програмно-технічному рівні на здійсненність процесів захисту інформації має вплив можливість закупівлі необхідних засобів захисту і технічні можливості їх застосування на об'єкті інформатизації, розмір фондів, що виділяються на реалізацію програми забезпечення безпеки інформації і планів захисту, наявність на ринку відповідних засобів захисту потрібної якості, технологічний рівень процесів обробки інформації на об'єкті інформатизації (стан парку обчислювальної та іншої спеціалізованої техніки, рівень комп'ютеризації та інформатизації технологічних процесів і т.ін.).

Враховуючи вищенаведене пропонується ввести до розгляду коефіцієнт здійсненності захисту інформації у вигляді

$$K_{pract}=F(I^{law}_{pract},I^{man}_{pract},I^{op}_{pract},I^{tech}_{pract}),$$

де I^{law}_{pract} – індекс, що характеризує здійсненність захисту інформації в організації на законодавчо-правовому рівні; I^{man}_{pract} – індекс, що характеризує здійсненність захисту інформації в організації S_0 на адміністративному рівні; I^{op}_{pract} – індекс, що характеризує здійсненність захисту інформації в організації S_0 на процедурному рівні; I^{tech}_{pract} – індекс, що характеризує здійсненність захисту інформації в організації S_0 на програмно-технічному рівні; F – оператор агрегування (формування узагальненого показника).

Таким чином, діяльність із захисту інформації є здійсненною, якщо на правовому, адміністративному, процедурному та програмно-

технічному рівнях забезпечення безпеки інформації створено всі умови для здійснення процесів захисту інформації.

8. ПРОСТОТА В АДМІНІСТРАТИВНОМУ ЗАБЕЗПЕЧЕННІ ЗАХИСТУ ІНФОРМАЦІЇ

Властивість *простоти* в адміністративному забезпеченні характеризує діяльність із захисту інформації з точки зору її придатності для адміністрування, тобто враховує наявність достатнього адміністративного ресурсу для реалізації процесів захисту інформації та впровадження політики безпеки, рівень професіоналізму, організаторських здібностей та навичок персоналу управління, що відповідає за організацію ефективного захисту інформації. Діяльність із захисту інформації є простою в адміністративному забезпеченні, якщо вона потребує мінімальних витрат на організаційно-штатні зміни в структурі організації.

ВИСНОВКИ

Таким чином, у рамках системодіяльнісної методології, захист інформації має бути цілеспрямованим, стабільним, безперервним, організованим, керованим, узгодженим та вмотивованим, забезпечувати максимальну ефективність, бути адекватним загрозам та ризикам безпеки, сприйнятим та гнучким в реалізації, здійсненним на різних рівнях забезпечення безпеки інформації, достатньо простим в адміністративному забезпеченні.

Сформульована у роботі множина властивостей не є вичерпною. Але її можна вважати ядром, на основі якого необхідно продовжувати дослідження щодо визначення аналітичних виразів відповідних показників та критеріїв оцінки властивостей, виявлення характеру взаємного впливу властивостей, розробки відповідного науково-методичного апарату оцінювання.

Література

[1] Надежность и эффективность в технике: Справочник: В 10 т./ Ред. Совет: В.С. Авдуевский и др. Т.3. Эффективность технических систем. – М.: Машиностроение, 1988 – 382 с.

[2] *Бинкин Б.А.* Эффективность управления: наука и практика. – М.:Наука, 1982.

[3] Эффективность торговли: сущность, измерение, оценка. – К.:Вища школа, 1986. – 32 с.

[4] *Маркіна І.А.* Методологічні питання ефективності управління // Фінанси України. – 2000. – №6. – С. 24-32.

[5] *Першиков В.И.*, *Савинков В.М.* Толковый словарь по информатике. – М.: Финансы и статистика, 1991.

[6] *Скрипник Л.В.*, *Потий А.В.* Гибкость и специализация профиля защиты автоматизированной системы. // Радиотехника. Всеукраинский межвед. Научн.-техн. Сб. – 2001. – Вып. 119. – С. 17-21.

Надійшла до редколегії 7.04.0.2012

Потій Олександр Володимирович, професор, доктор техн. наук, начальник кафедри радіоелектронних систем пунктів управління Повітряних Сил ім. І. Кожедуба. Область наукових інтересів: проектування комплексних систем захисту інформації, системний аналіз процесів захисту інформації, управління захистом інформації.

Пилипенко Дмитро Юрійович, аспірант кафедри БІТ ХНУРЕ. Область наукових інтересів: управління процесами захисту інформації.

Гладкий Дмитро Іванович, молодший науковий співробітник науково-дослідної лабораторії системних технологій ХНУРЕ. Область наукових інтересів: аналіз ризиків, проблеми забезпечення безпеки інформації в ІТС.

УДК 681.3.06

Свойства деятельности по обеспечению защиты информации как системной категории / А.В. Потий, Д.Ю. Пилипенко, Д.И. Гладкий // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 299–303.

В работе исследуются свойства деятельностного аспекта защиты информации как системной категории. Сформированное множество свойств позволяет проводить всестороннюю оценку деятельности по обеспечению защиты информации. Данное множество свойств представляет определенную основу для дальнейших исследований и разработки аналитических выражений и критериев оценки системы защиты информации в организации.

Ключевые слова: системодеятельностный подход, деятельность по обеспечению защиты информации, оценивание.

Ил. 02. Библиогр.: 06 назв.

UDC 681.3.06

The properties of information security activities as a system category / A.V. Potiy, D.Yu. Pilipenko, D.I. Gladkiy // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 299–303.

The properties of information security activities as a system category are studied. The proposed set of properties enables a comprehensive evaluation of information security activities. The given set of properties can be considered as the basis for further research and development of analytical forms and criteria for IS system evaluation in an organization.

Keywords: system-structural approach, information security activities, evaluation.

Fig. 02. Ref.: 06 items.

УДК 621.34

МЕТОД ОЦЕНКИ РИСКОВ НАРУШЕНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ БАНКОВСКИХ УЧРЕЖДЕНИЙ

Д.В. СЕМЁНОВ, Ф.Л. ДЕМЧЕНКО

Предлагается метод оценки рисков нарушения информационной безопасности на основе опроса кадрового состава предприятия по трем направлениям: текущий уровень информационной безопасности организации; оценка системы управления информационной безопасностью организации; оценка уровня осознания руководством необходимости обеспечения информационной безопасности организации

Ключевые слова: критерий безопасности, оценка рисков, безопасность банковских учреждений.

ВВЕДЕНИЕ

Работа банковских систем вплотную связана с необходимостью постоянно держать в безопасности циркулирующую там информацию. Если говорить о банковской системе, определенно речь идет о крупномасштабной инфраструктуре, каждый элемент которой нуждается в определенной степени защиты. Однако максимально возможная защита всей системы является чрезмерно дорогостоящим мероприятием, и тут встает вопрос оценки рисков. Оценка рисков представляет собой комплексное изучение системы. Результатом оценки рисков является итоговый уровень информационной безопасности организации.

Методика оценки соответствия информационной безопасности организаций банковской системы используется для проведения аудита и оценки уровня обеспечения информационной безопасности организаций, осуществляющих деятельность в банковской сфере [1]. Целью методики является проведение оценки рисков информационной безопасности по направлениям:

- текущий уровень информационной безопасности организации($EV1$);
- менеджмент информационной безопасности организации($EV2$);
- уровень осознания информационной безопасности организации($EV3$).

1. ГРУППОВЫЕ И ЧАСТНЫЕ ПОКАЗАТЕЛИ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ. СПОСОБЫ ОЦЕНИВАНИЯ ПОКАЗАТЕЛЕЙ

Для оценки информационной безопасности организации используются групповые и частные показатели [2]. Групповые показатели образуют структуру направлений оценки. Оценки групповых показателей (EV_{Mi}) используются для получения оценки по направлениям ($EV1$, $EV2$ и $EV3$). Частные показатели входят в состав групповых показателей и представлены в виде вопросов, ответы на которые дают возможность определить оценки (EV_{Mij}), которые затем формируют оценки EV_{Mi} групповых показателей.

Для проведения оценки используются формы, содержащие вопросы, групповой показатель, входящие в него частные показатели и их категории (обязательность выполнения), метрику (шкалу) для оценивания частных показателей и коэффициенты значимости частных показателей, используемые при вычислении группового показателя.

Оценка EV_{Mij} частного показателя формируется на основании выявленной аудиторской группой степени выполнения требований посредством экспертного оценивания.

При проведении оценки частных показателей используется шкала оценивания, представленная в табл. 1.1, 1.2, 1.3.

Таблица 1.1

Критерии оценки частных показателей для оценки «степени документированности» и «выполнения требований информационной безопасности»

Оценка частного показателя	Критерий выставления оценки частного показателя
0	Требования частного показателя не установлены во внутренних нормативных документах проверяемой организации и не выполняются
0	Требования частного показателя частично установлены в нормативных документах проверяемой организации, но не выполняются
0,25	Требования частного показателя полностью установлены в нормативных документах проверяемой организации, но не выполняются
0,25	Требования частного показателя не установлены во внутренних нормативных документах проверяемой организации и выполняются в неполном объеме
0,25	Требования частного показателя частично установлены во внутренних нормативных документах проверяемой организации и выполняются в неполном объеме
0,5	Требования частного показателя полностью установлены во внутренних нормативных документах проверяемой организации и выполняются в неполном объеме
0,5	Требования частного показателя не установлены во внутренних нормативных документах проверяемой организации, но выполняются в полном объеме

0,75	Требования частного показателя частично установлены во внутренних нормативных документах проверяемой организации, но выполняются в полном объеме
1	Требования частного показателя полностью установлены во внутренних нормативных документах проверяемой организации и выполняются в полном объеме

Таблица 1.2

Критерии выставления оценок частных показателей для оценки «степени документированности требований информационной безопасности»

Оценка частного показателя	Критерий выставления оценки частного показателя
0	Требования частного показателя не установлены во внутренних нормативных документах проверяемой организации
0,5	Требования частного показателя частично установлены в нормативных документах проверяемой организации
1	Требования частного показателя полностью установлены в нормативных документах проверяемой организации

Таблица 1.3

Критерии выставления оценок частных показателей для оценки «степени выполнения требований информационной безопасности»

Оценка частного показателя	Критерий выставления оценки частного показателя
0	Требования частного показателя не выполняются
0,5	Требования частного показателя выполняются в неполном объеме
1	Требования частного показателя выполняются в полном объеме

2. ОЦЕНКА ТЕКУЩЕГО УРОВНЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ОРГАНИЗАЦИИ

Оценка текущего уровня информационной безопасности организации определяется совокупностью групповых показателей, позволяющих оценить степень выполнения требований (табл. 2).

Таблица 2

Перечень требований для оценки текущего уровня ИБ организации

Обозначение группового показателя	Наименование группового показателя (требование)
М1	Обеспечение ИБ при назначении и распределении ролей и обеспечении доверия к персоналу
М2	Обеспечение ИБ на стадиях жизненного цикла АС
М3	Обеспечение ИБ при управлении доступом и регистрацией
М4	Обеспечение ИБ средствами антивирусной защиты
М5	Обеспечение ИБ при использовании ресурсов сети Интернет
М6	Обеспечение ИБ при использовании средств криптографической защиты информации
М7	Обеспечение ИБ банковских платежных технологических процессов
М8	Обеспечение ИБ банковских информационных технологических процессов
М9	Обработка персональных данных в организации
М10	Обеспечение ИБ банковских технологических процессов, в рамках которых обрабатываются персональные данные

Оценка группового показателя текущего уровня ИБ организации (EV_{Mi}) вычисляется из оценок входящих в него частных показателей (EV_{Mij}) с учетом коэффициентов значимости α_{ij}, определяющих важность частного показателя для оценивания группового показателя:

$$EV_{Mi} = \sum_j \alpha_{ij} * EV_{Mij} \,. \qquad (1)$$

При формировании коэффициентов значимости учитывается следующее условие нормировки:

$$\sum_{j=1}^{k} \alpha_{ij} = 1 \,, \qquad (2)$$

где k — число частных показателей в i-м групповом показателе.

Оценка степени выполнения требований безопасности, регламентирующих банковский информационный технологический процесс $EV_{\text{БИТП}}$ вычисляется по формуле принимая во внимание результаты оценивания групповых показателей $M1$-$M6$:

$$EV_{\text{БИТП}} = \frac{\sum_i EV_{Mi} + EV_{M8}}{7}, i = 1 \div 6 \qquad (3)$$

Оценка степени выполнения требований безопасности, регламентирующих защиту персональных данных в информационных системах персональных данных $EV^1_{\text{ОЗПД}}$, без учета требований при использовании средств криптографической защиты информации вычисляется по формуле:

$$EV^1 = \frac{\sum_i EV_{Mi} + EV_{M8} + EV_{M10}}{7}, i = 1 \div 5 \,. \qquad (4)$$

Оценка степени выполнения требований, регламентирующих защиту персональных данных $EV^2_{\text{ОЗПД}}$, с учетом оценки степени выполнения требований при использовании криптографической защиты информации вычисляется по формуле:

$$EV^2_{\text{ОЗПД}} = \frac{\sum_i EV_{Mi} + EV_{M8} + EV_{M10}}{8}, i = 1 \div 6 \,. \qquad (5)$$

3. ОЦЕНКА МЕНЕДЖМЕНТА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ОРГАНИЗАЦИИ

Организация должна вводить, выполнять, использовать, контролировать, пересматривать, поддерживать и совершенствовать документированные положения системы управления информационной безопасностью в рамках всей бизнес-деятельности организации, а также рисков, с которыми она сталкивается.

Данные требования, обеспечиваются с помощью процесса, который основывается на модели «Планирование – реализация – оценка - корректировка» (рис. 1).

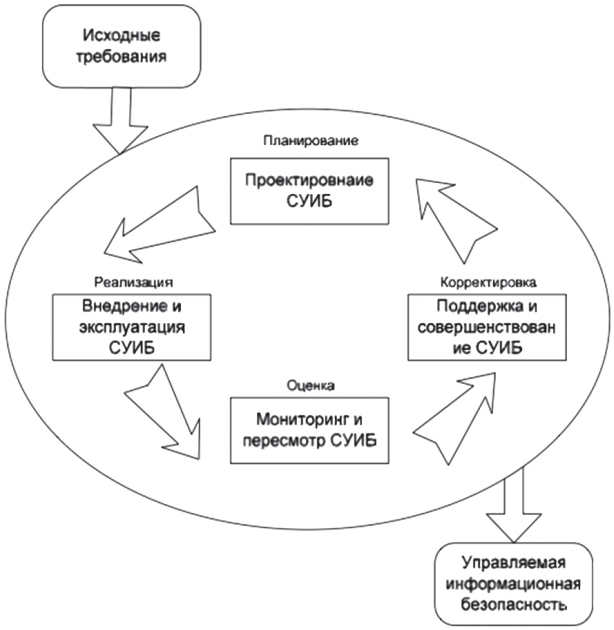


Рис. 1. Модель «Планирование – реализация – оценка – корректировка»

Оценка менеджмента информационной безопасности организации определяется совокупностью групповых показателей, позволяющих оценить степень соблюдения данных требований.

Таблица 3

Перечень требований для оценки менеджмента ИБ организации

Обозначение группового показателя	Наименование группового показателя (требование)
M11	Организация и функционирование службы ИБ организации
M12	Определение/коррекция области действия системы обеспечения информационной безопасности (СОИБ)
M13	Выбор/коррекция подхода к оценке рисков нарушения ИБ и проведение оценки рисков нарушения ИБ
M14	Разработка планов обработки рисков нарушения ИБ
M15	Разработка/коррекция внутренних документов, регламентирующих деятельность в области обеспечения ИБ
M16	Принятие руководством организации решений о реализации и эксплуатации СОИБ
M17	Организация реализации планов внедрения СОИБ
M18	Разработка и организация реализации программ по обучению и повышению осведомленности в области ИБ
M19	Организация обнаружения и реагирования на инциденты безопасности
M20	Организация обеспечения непрерывности бизнеса и его восстановления после прерываний
M21	Мониторинг и контроль защитных мер
M22	Проведение самооценки ИБ
M23	Проведение аудита ИБ
M24	Анализ функционирования СОИБ
M25	Анализ СОИБ со стороны руководства организации
M26	Принятие решений по тактическим улучшениям СОИБ
M27	Принятие решений по стратегическим улучшениям СОИБ

Итоговая оценка $EV2$, отражающая степень выполнения требований по направлению “менеджмент информационной безопасности организации”, вычисляется по формуле:

$$EV2 = \frac{\sum_{i=11}^{27} EV_{Mi}}{17}. \qquad (6)$$

4. ОЦЕНКА УРОВНЯ ОСОЗНАНИЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ОРГАНИЗАЦИИ

Оценка уровня осознания информационной безопасности организации определяется совокупностью групповых показателей, позволяющих оценить степень выполнения требований для следующих областей (см. табл. 4).

Таблица 4

Перечень требований для оценки уровня осознания ИБ организации

Обозначение группового показателя	Наименование группового показателя (требование)
M28	Оценка деятельности руководства организации по поддержке функционирования службы ИБ организации
M29	Оценка деятельности руководства организации по принятию решений о реализации и эксплуатации СОИБ
M30	Оценка деятельности руководства организации по поддержке планирования СОИБ
M31	Оценка деятельности руководства организации по поддержке реализации СОИБ
M32	Оценка деятельности руководства организации по поддержке проверки СОИБ

M33	Оценка деятельности руководства организации по анализу СОИБ
M34	Оценка деятельности руководства организации по поддержке совершенствования СОИБ

Итоговая оценка EV3, отражающая степень выполнения требований по направлению “уровень осознания ИБ организации”, вычисляется по формуле:

$$EV3 = \frac{\sum_{i=28}^{34} EV_{Mi}}{7}. \quad (7)$$

5. ОПРЕДЕЛЕНИЕ УРОВНЯ ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ БАНКОВСКОЙ ОРГАНИЗАЦИИ. ОТОБРАЖЕНИЕ ОЦЕНОК

Если оценка *EV*1, *EV*2 или *EV*3 лежит в интервале от 0 до 0,25, то данному направлению оценки присваивается нулевой уровень обеспечения информационной безопасности.

Если оценка *EV*1, *EV*2 или *EV*3 лежит в интервале от 0,25 до 0,5, то данному направлению оценки присваивается первый уровень обеспечения информационной безопасности.

Если оценка *EV*1, *EV*2 или *EV*3 лежит в интервале от 0,5 до 0,7, то данному направлению оценки присваивается второй уровень обеспечения информационной безопасности.

Если оценка *EV*1, *EV*2 или *EV*3 лежит в интервале от 0,7 до 0,85, то данному направлению оценки присваивается третий уровень обеспечения информационной безопасности.

Если оценка *EV*1, *EV*2 или *EV*3 лежит в интервале от 0,85 до 0,95, то данному направлению оценки присваивается четвертый уровень обеспечения информационной безопасности.

Если оценка *EV*1, *EV*2 или *EV*3 лежит в интервале от 0,95 до 1 включительно, то данному направлению оценки присваивается пятый уровень обеспечения информационной безопасности.

Итоговый уровень обеспечения информационной безопасности - Значение *R* определяется по наименьшему значению из трех оценок по направлениям оценки:

- оценки уровня осознания информационной безопасности организации (*EV*3);
- оценки менеджмента информационной безопасности организации (*EV*2);
- оценки текущего уровня информационной безопасности организации (*EV*1).

Полученное в результате оценки соответствия организации уровню обеспечения информационной безопасности, значение *R* является основой для формирования аудиторского заключения по результатам аудита информационной безопасности.

Чем больше значение *R*, тем соответствующий этому значению уровень обеспечения информационной безопасности организации выше.

Для отображения результатов оценивания используется круговая диаграмма (см. рис. 3.2).

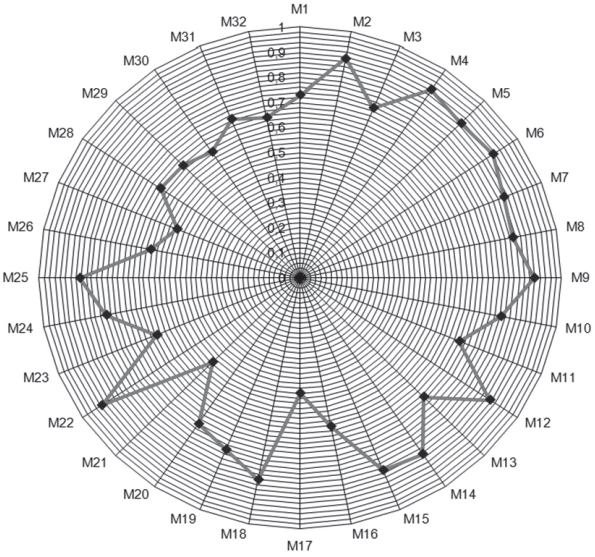


Рис. 2. Круговая диаграмма для отображения результатов оценивания

Секторы с 1-го по 10-й используются для отображения оценки текущего уровня обеспечения информационной безопасности организации.

Секторы с 11-го по 27-й используются для отображения оценки процессов менеджмента информационной безопасности организации.

Секторы с 28-го по 34-й используются для отображения оценки уровня осознания информационной безопасности организации.

ЗАКЛЮЧЕНИЕ

В данной статье предложен метод оценки состояния информационной безопасности банковской организации. Оценка уровня информационной безопасности проводилась по трем направлениям: текущий уровень информационной безопасности организации; оценка системы управления информационной безопасностью организации; оценка уровня осознания руководством необходимости обеспечения информационной безопасности организации. Проведенные работы показали, что целесообразным является проведение оценки не только текущего уровня информационной безопасности организации, но и проведение оценки осознания необходимости обеспечения информационной безопасности, а также оценки системы управления информационной безопасностью организации. Предлагаемая методика оценки рисков характеризуется высокой гибкостью применения и глубиной раскрытия, как частных, так и групповых показателей информационной безопасности. Данная методика может быть использована специалистами, проводящими аудит информационной безопасности банковских учреждений.

В целом рассмотренные в данной статье вопросы позволяют оценить уровень текущего состояния защищенности информационных

ресурсов предприятия, а также выработать рекомендации по обеспечению (повышению) информационной безопасности, в том числе снизить потенциальные потери предприятия путем повышения устойчивости функционирования, автоматизированной системы, разработать концепцию и политику безопасности компании. Предлагаемая методика оценки рисков информационной безопасности банковской организации позволяет сформировать меры защиты конфиденциальной информации компании.

Литература

[1] *С.А. Петренко, С.В. Симонов.* Анализ и управление информационными рисками. – ДМИ Пресс, 2004.

[2] *Белкин А.Р., Левин М.Ш.* Принятие решений: комбинаторные модели апроксимации информации. – М.: Наука, 1990.

Поступила в редколлегию 17.04.2012

Семёнов Дмитрий Владимирович, магистрант кафедры БИТ ХНУРЭ. Область научных интересов: управление информационной безопасностью.

Демченко Фёдор Леонидович, магистрант кафедры БИТ ХНУРЭ. Область научных интересов: защита информации в информационно-телекоммуникационных системах.

УДК 621.34

Метод оцінки ризиків порушення інформаційної безпеки банківських закладів / С.Г. Семенов, Ф.Л. Демченко // Прикладна радіоелектроніка: наук.-техн. журнал. – 2012. – Том 11. № 2. – С. 304–308.

Пропонується метод оцінки ризиків порушення інформаційної безпеки на основі опитування кадрового складу підприємства за трьома напрямками: поточний рівень інформаційної безпеки організації; оцінка системи управління інформаційною безпекою організації; оцінка рівня усвідомлення керівництвом необхідності забезпечення інформаційної безпеки організації.

Ключові слова: критерій безпеки, оцінка ризиків, безпека банківських установ.

Табл. 06. Іл. 02. Бібліогр.: 2 найм.

UDC 621.34

Method of evaluating risks of violating information security of banking establishments / D.V. Semenov, F.L. Demchenko // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 304–308.

The paper suggests a method of evaluating risks of information security violation on the basis of questioning an enterprise's personnel in three directions: current level of an organization's information security; evaluating a control system of an organization's information security; evaluating the level of the governing body's realization of the need for ensuring the organization's info security.

Keywords: security criterion, evaluation of risks, security of banking establishments.

Tab. 06. Fig. 06. Ref.: 2 items.

УДК 621.34

БІОМЕТРИЧНА АУТЕНТИФІКАЦІЯ НА ОСНОВІ АНАЛІЗУ КЛАВІАТУРНОГО ПОЧЕРКУ

С.О. ЄНГАЛИЧЕВ, С.Г. СЕМЕНОВ

У статті проведена оцінка стабільності клавіатурного почерку користувача. Розроблена імітаційна модель системи автентифікації користувачів на основі клавіатурного почерку. Проведені дослідження і виявлені закономірності в клавіатурному почерку різних користувачів.

Ключові слова: біометричні системи автентифікації, клавіатурний почерк, імітаційна модель.

Актуальність. Аналіз сучасного українського ринку технічних засобів забезпечення безпеки показав, що, в розвитку індустрії безпеки сьогодні позначився новий етап. На загальному фоні ринку, що стабілізувався, найдинамічніше продовжують розвиватися сучасні системи автентифікації особи і захисту інформації. Особливу увагу привертають до себе біометричні засоби захисту інформації (БСЗІ), що пояснюється їх високою надійністю автентифікації і досягнутим останнім часом значним зниженням їх вартості.

В теперішній час існує безліч методів біометричної автентифікації користувачів комп'ютерних систем, які можна розділити на дві великі групи: статичні і динамічні [1-4]. Класифікація методів біометричної аутентифікації представлена на рис. 1.

Рис. 1. Класифікація методів біометричної автентифікації користувачів

Аналіз методів біометричної автентифікації показав, що статичні методи, грунтовані на характеристиці людини, тобто унікальній властивості, даній йому від народження і невід'ємному від нього, разом з достоїнствами (висока точність автентифікації, висока швидкість реакції та ін.) мають ряд недоліків (висока вартість устаткування, витратних матеріалів і обслуговування) [1, 3]. Методи динамічної автентифікації грунтуються на поведінковій (динамічній) характеристиці людини, т. е. побудовані на особливостях, характерних для підсвідомих рухів в процесі відтворення якої-небудь дії. Аналіз методів динамічної автентифікації показав, що, незважаючи на ряд їх недоліків (висока імовірність помилки автентифікації і неправдивих спрацьовувань системи безпеки), в силу простоти і доступності вони залишаються затребуваними в секторі невеликих установ, підприємств і організацій [1, 2].

Аналіз джерел [1-4] показав, що серед динамічних методів біометричної автентифікації ряд авторів виділяє методи автентифікації користувача по клавіатурному почерку. Пов'язано це в основному з тим, що клавіатурний почерк користувача має певну стабільність, що дозволяє з високою імовірністю правильно автентифіківати користувача, працюючого з клавіатурою.

Основна частина. Проведені дослідження показали, що автентифікація користувачів в цих методах, як правило, грунтується на статистичних методах обробки початкових даних і формуванні вихідного вектору, який є ідентифікатором цього користувача. В якості початкових даних використовують часові інтервали між натисненням клавіш на клавіатурі і час їх утримання. При цьому часові інтервали між натисненням клавіш характеризують темп роботи, а час утримання клавіш характеризує стиль роботи з клавіатурою – різкий удар або плавне натиснення.

Дослідження показали, що автентифікація користувача по клавіатурному почерку можлива наступними способами: по набору ключової фрази; по набору довільного тексту.

Обидва способи мають на увазі два режими роботи: навчання; автентифікація.

На етапі навчання користувач вводить деяке число раз пропоновані йому тестові фрази. При цьому розраховуються і запам'ятовуються еталонні характеристики цього користувача. На етапі автентифікації розраховані оцінки порівнюються з еталонними, на підставі чого робиться висновок про збіг або неспівпадання параметрів клавіатурного почерку.

Еталонні характеристики користувача, отримані на етапі навчання системи, дозволяють зробити висновки про міру стабільності клавіатурного почерку користувача і визначити довірчий інтервал розкиду параметрів для подальшої автентифікації користувача. Щоб уникнути дискредитації роботи системи можливо відсівати користувачів, клавіатурний почерк яких не має необхідної «стабільності» [1, 2, 4]. Для цього можна користуватися даними, представленими в табл.1.

Таблиця 1

Оценка стабильности клавиатурного почерка пользователя

Помилки,%	Аритмічність, %	Швидкість, зн./мін	Характеристика перекриттів		
			Число перекриттів, %	Використовуване число пальців	Оцінка
менш 2	менш 10	більше 200	більше 50	все	відмінно
менш 4	менш 15	більше 150	більше 30	більшість	добре
менш 8	менш 20	більше 100	більше 10	декілька	задов.
більше 8	більше 20	менш 100	менш 10	по одному	неуд.

Розробимо імітаційну модель клавіатурного введення інформації і проведемо на її основі дослідження біометричних закономірностей користувачів.

Імітаційна модель дозволяє аналізувати вектори часів утримань клавіш, а так само інтервалів часу між натисненнями клавіш.

Експеримент полягає у введенні з клавіатури спеціально підібраної кількості слів і словосполучень з реєстрацією часів утримання клавішею для покриття усього вектору часів утримань клавіш, а також з реєстрацією часів міжсимвольних інтервалів, які фіксуються у відповідній матриці. Цей набір слів кожен користувач набирає в різний час доби (вдень і уночі).

Проаналізуємо дві найбільш типових біометричних характеристики двох користувачів, які в цілому відбивають основні особливості роботи користувачів в різний час доби.

Динаміка зміни середнього часу утримання клавіш користувачами в різний час доби представлена на рис. 2.

Аналіз кривих зміни середнього часу утримання клавіш користувачами в різний час доби показав незначне (до 1,02 разів) зменшення часу натиснення клавіш в середини робочого дня, і збільшення цього часу до 1,05 разу о 5.00 годині ранку при безперервній роботі в нічний час.

Криві зміни середніх інтервалів часу між натисненнями клавіш в різний час доби представлені на рис. 3. (а – перший користувач, б – другий користувач).

Криві рис. 3. так само ілюструють результати процесу аутентифікації користувача по різниці змін середніх інтервалів часу між натисненнями клавіш (крива Average). Як можна помітити з рисунку висунене припущення про стабільність клавіатурного почерку користувача в цілому підтверджується (відхилення даних верифікацій легального користувача тільки в 5% випадків перевищують 70%). Приведені характеристики говорять також про ритм роботи кожного користувача з клавіатурою. Окрім цього, можна виявити і ще одну важливу закономірність. У ряді випадків абстрактний користувач (тобто деякий образ усередненого користувача) стабільніше працює вдень в порівнянні з нічним часом доби. Так в прикладі, представленому на рис. 3. перший користувач «стабільніше» працює вдень майже в 1,3 разу, другий користувач в 1,051 рази. Дослідження показали, що і цей факт необхідно використати проектувальникам при розробці систем біометричної автентифікації.

Висновки. На основі імітаційної моделі системи біометричної автентифікації користувачів були проведені дослідження клавіатурного почерку окремих користувачів в денний і нічний час доби. Були виявлені ряд закономірностей, що підтверджують гіпотезу про можливість динамічної біометричної автентифікації користувачів по клавіатурному почерку. Крім того, результати досліджень показали можливість використання запропонованої моделі в системах захисту інформаційно-обчислювальних комплексів від несанкціонованого доступу.

Література

[1] *Брюхомицкий Ю.А.* Исследование биометрических систем динамической аутентификации пользова-

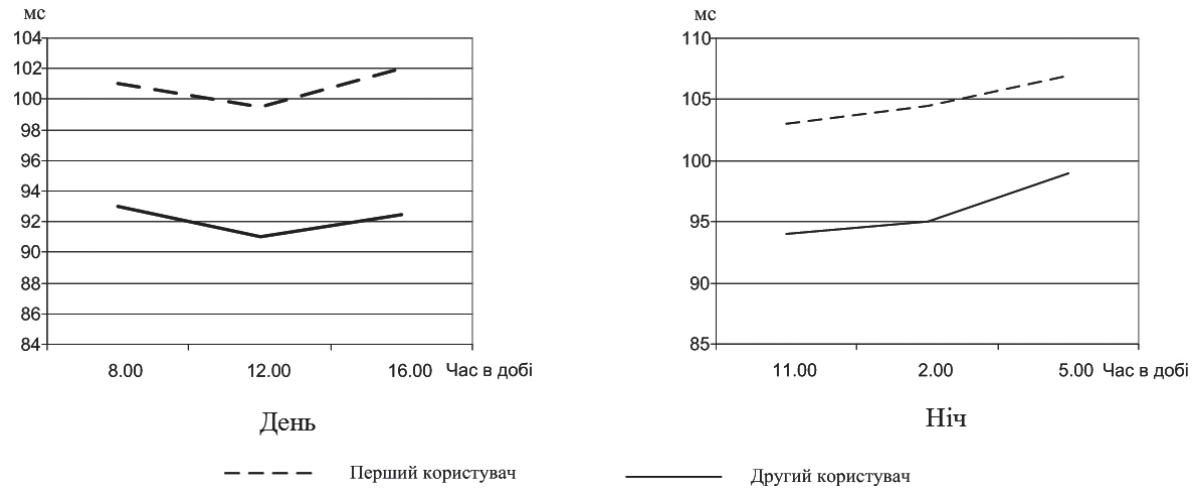
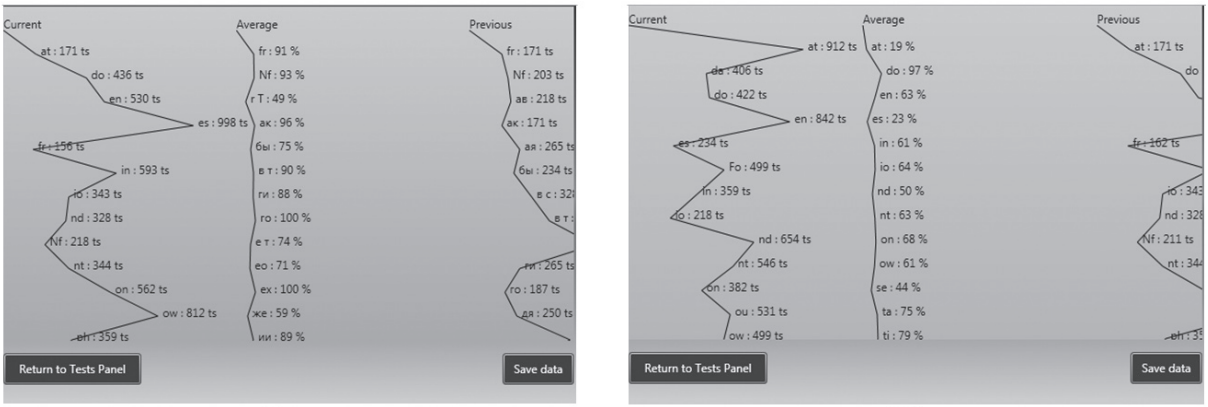
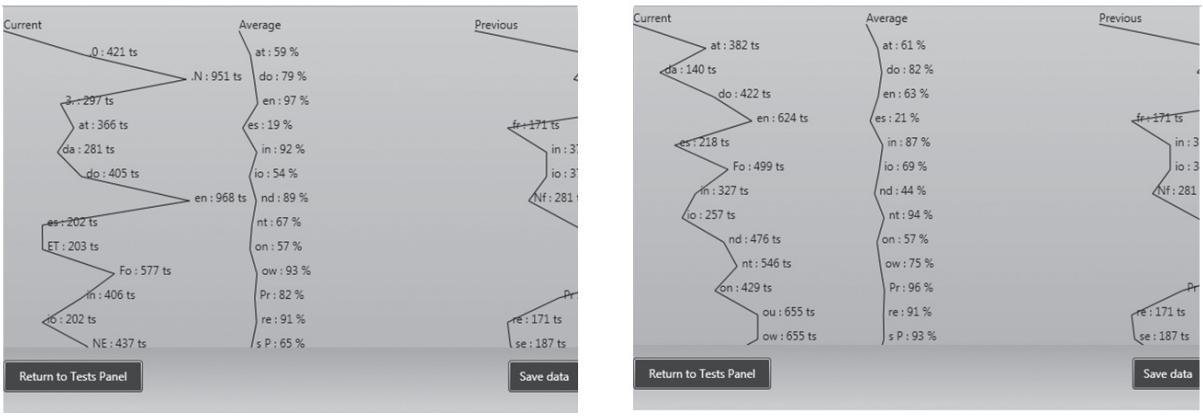


Рис. 2. Криві зміни середнього часу утримання клавіш користувачами в різний час доби



День Ніч

а



День Ніч

б

Рис. 3. Криві зміни середніх інтервалів часу між натисненнями клавіш в різний час доби

телей ПК по рукописному и клавиатурному почеркам / Ю.А. Брюхомицкий, М.Н. Казарин – Таганрог: Изд-во ТРТУ, 2004. – 38 с.

[2] *Кобиелус Д.* Информационная безопасность: идентификация и аутентификация / Джеймс Кобиелус – М.: Связь, 1997. – 252 с.

[3] *Карабутов Н.Н.* Структурная идентификация систем: анализ динамических структур / Н.Н.Карабутов. – М.: МГИУ, 2008. – 160 с.

[4] *Расторгуев С.П.* Программные методы защиты информации в компьютерах и сетях / С.П. Расторгуев – М.: Изд-во Агентства «Яхтсмен», 1993. 188 с.

Поступила в редколлегию 24.04.2012

Семенов Сергій Геннадійович, к.т.н. доцент кафедри обчислювальної техніки і програмування, Національний Технічний університет "ХПІ". Галузь наукових інтересів: Захист інформації в інформаційно-телекомунікаційних системах.

Єнгаличев Сергій Олександрович, магістр, Харківський національний університет радіоелектроніки. Галузь наукових інтересів: Захист інформації в комп'ютерних системах.

УДК 621.34

Биометрическая аутентификация на основе анализа клавиатурного почерка / С.А.Енгалычев, С.Г. Семенов // Прикладная радиоэлектроника: науч.-техн. журнал. – 2012. – Том 11. № 2. – С. 309–311.

В статье проведенная оценка стабильности клавиатурного почерка пользователя. Разработана имитационная модель системы аутентификации пользователей на основе клавиатурного почерка. Проведенные исследования и выявленные закономерности в клавиатурном почерке различных пользователей.

Ключевые слова: биометрические системы аутентификации, клавиатурный почерк, имитационная модель

Табл. 01. Ил.02. Библиогр.: 04 назв.

UDC 621.34

Biometric authentication on the basis of analysis of keyboard handwriting / S.A. Engalychev, S.G. Semenov // Applied Radio Electronics: Sci. Journ. – 2012. Vol. 11. № 2. – P. 309–311.

Estimation of the stability of a user keyboard handwriting is conducted in the paper. A simulation model of a user authentication system on the keyboard handwriting basis is developed. Research is done and regularities in different users' keyboard handwriting are revealed.

Keywords: biometric systems of authentication, keyboard handwriting, simulation model.

Tab. 01. Fig. 03. Ref.: 04 items.

ПРИКЛАДНАЯ РАДИОЭЛЕКТРОНИКА

Научно-технический журнал

Ответственный секретарь

Е. Б. Исаева

Корректор

А. И. Шахова

Перевод на английский язык

К. Т. Умяров

Компьютерный дизайн и верстка

Е. Б. Исаева

Рекомендовано засіданням Бюро Президії Академії наук прикладної радіоелектроніки
(протокол № 2 від 27.06.2012 р.).

Рекомендовано Вченою радою Харківського національного університету радіоелектроніки
(протокол № 13 від 8.06.2012 р.).

Свідоцтво про державну реєстрацію КВ № 6037 від 09.04.2002 р.

Журнал включений у список фахових видань ВАК України
по технічним наукам
(постанова президії ВАК України № 1-05/2 от 10.03.2010),
по фізико-математичним наукам (фізика)
(постанова президії ВАК України № 1-05/5 от 1.07.2010)

Підписано до друку 27.06.2012. Формат 60 × 84 $^{1}/_{8}$.
Папір офсет. Друк офсет. Умов.-друк. арк. 21,9. Облік.-вид. арк. 22,0.
Тираж 300 прим. Ціна договірна.

Віддруковано в ТОВ «ДРУКАРНЯ МАДРИД»
61024, м. Харків, вул. Ольмінського, 8. Тел.: (057) 717-41-79
www.madrid.in.ua, e-mail: info@madrid.in.ua