УДК 004.056.55

В.А. Лужецький, О.В. Дмитришин

Вінницький національний технічний університет, Вінниця

ВИКОРИСТАННЯ ОПЕРАЦІЇ МНОЖЕННЯ ЗА МОДУЛЕМ В СИМЕТРИЧНИХ БЛОКОВИХ ШИФРАХ

Розглянуто один із методів використання операції множення за модулем в блокових шифрах. Запропоновано методи генерування взаємно простих чисел та пошуку обернено мультиплікативних.

Ключові слова: *блоковий шифр, множення за модулем, взаємно прості числа, обернено мультиплікативні.*

Вступ

В криптографії, під час розробки симетричних блокових шифрів, використовують різні базові криптоперетворення, які за характером реалізації можна розділити на складні та елементарні. До складних криптографічних перетворень відносяться операції перестановок і підстановок. До елементарних – логічні та побітові операції, операції додавання та множення за модулем. Серед двох вище згаданих груп операцій, саме операція множення є найбільш універсальною операцією, яка реалізована в більшості сучасних мікропроцесорах. Кожен біт результату множення здебільшого нелінійним чином [1] залежить від усіх бітів перетворюваного блоку даних.

Проте, використання операції множення за модулем спричинює деякі складнощі під час розробки симетричних блокових шифрів, зокрема це пошук оберненого мультиплікативного елемента [2 – 4].

Метою даної роботи є підвищення стійкості блокових шифрів до диференційного криптоаналізу, що використовують операцію множення за модулем.

Постановка задач досліджень:

1. Проаналізувати відомі підходи проектування блокових шифрів, що використовують операцію множення за модулем.

2. Розробити підхід, щодо використання операції множення за невідомим модулем, який забезпечує стійкість до диференційного криптоаналізу на основі мультиплікативних диференціалів.

Відомі підходи до використання операції множення в блокових шифрах

Одним із перших симетричних блокових шифрів, який використав операцію множення за модулем є шифр MMB (Modular multiplication based block cipher) [5], з розміром секретного ключа та блока даних в 128 біт. Операція множення за модулем здійснюється над 32-бітовими підблоками даних за таким правилом:

$$X \times A = \begin{cases} X \cdot A \bmod (2^{32}-1), \text{ якщо } X < 2^{32}-1; \\ 2^{32}-1, \text{ якщо } X = 2^{32}-1, \end{cases}$$

де X – вхідний блок даних; A – множник.

При цьому, під час розшифрування має існувати таке A^{-1}, що $A \cdot A^{-1} = 1 \bmod 2^{32}-1$. Операція множення, в шифрі MMB, використана фактично як деяка модифікована операція інверсії, оскільки множники A та A^{-1} є константами, а модуль є відкритим.

Наступний симетричний блоковий шифр, який привертає нашу увагу є шифр ABC [6], що використовує 256 бітні блоки даних та 512 бітний ключ. Він, так само як і попередній шифр, виконує операцію множення над 32-бітними блоками даних із використанням відкритого модуля і в якості множників ті ж самі константи.

Розробники шифрів MMB та ABC використали операцію множення за модулем доволі обмежено, лише як деяку функцію побітового перемішування, не використовуючи всю повноту нелінійного перетворення бітів вхідного блока даних.

Варто відзначити, що в таких шифрах як MARS [1], Xenon [7] та RC6[8] з довжиною оброблюваного блока даних 128 біт і змінною довжиною ключа від 128 біт, операція множення використовується лише як допоміжна операція. Множення виконується за модулем 2^{32} і множник A повинен бути непарним числом. Обернені мультиплікативні для даних шифрів не потрібні.

Симетричний блоковий шифр xmx [9], розроблено виключно на операції множення та операції виключного-АБО. Зашифруваня блока даних виконується L раундів і полягає в побітовому додаванні із секретним ключем A за модулем 2, отриманий результат множать на теж саме значення A за модулем $m = 2^n-1$ (де n =512, 768, 1024). Розшифрування виконують в оберненому порядку і в якості множника використано обернене мультиплікативне A за модулем m, тобто $A^{-1} \bmod m$, яке обчислюється за допомогою розширеного алгоритму Евкліда.

Хоча розробники шифру використали дві групи несумісних операцій, проте використання одного й того самого підключа, для двох різних груп операцій, на всіх раундах шифрування і відкрите значення модуля не забезпечило достатню криптографічну стійкість шифру [10].

Операцію множення за модулем 2^{32} використовує блоковий шифр MultiSwap [11], що має розмір секретного ключа 374 біт і розмір оброблюваних блоків даних 64 біт. У цьому шифрі, використовується 12 підключів із яких 10 підключів використано для операції множення, наймолодший біт яких встановлений в одиницю і 2 підкюча для операцій додавання за модулем 2^{32}. Після кожної операції множення за модулем 2^{32} виконується перестановка старших та молодших 16 біт оброблюваного блоку між собою. Зашифрування даних складається з 12 раундів, з яких 5 та 11 раунди полягають у додаванні за модулем 2^{32}, а всі інші складаються з операцій множення та перестановки. Розробник шифру не використовував сам шифр для шифрування, а лише для створення MAC-коду повідомлення. Проте, щоб розшифрувати зашифроване повідомлення достатньо знайти обернені мультиплікативні за модулем 2^{32} для операції множення.

Недоліком даного шифру як і попереднього є його нестійкість до диференційного криптоаналізу за рахунок відомого значення модуля [10].

Автор роботи [2], в симетричному блоковому шифрі Nimbus з розміром секретного ключа в 128 біт, а пізніше і в шифрі Caligo [3] з змінним розміром ключа, що залежить від розміру блоку даних, пропонує виконувати множення за n–бітним модулем $m = 2^n$ (для шифру Nimbus n = 64, а для шифру Caligo – n = 128, 256, 320, 512):

$$Y = X \cdot A \bmod 2^n,$$

де A – n-бітне непарне число; Y – n-бітний вихідний блок даних.

Формування множників A, в даних шифрах, відбувається за рахунок використання початкових констант і n-бітного початкового секретного ключа та їх перетворення. Загальна кількість секретних підключів, яка може бути використана для операції множення дорівнює 2^{n-1}.

Процедура пошуку значень обернено мультиплікативних полягає у розв'язанні такого рівняння [2]:

$$\varphi\left(A, m^2\right) = \left(2 - A \cdot \varphi(r, m)\right) \cdot \varphi(r, m), \qquad (1)$$

де $r \equiv A \bmod m$; $\varphi(r, m)$ – обернене мультиплікативне r в полі Z_m.

Для n = 64 складність розв'язання рівняння (1) потребує виконання 6 редукцій.

Сама процедура зашифрування в блоковому шифрі Nimbus виконується таким чином:

$$Y_i = \left(Y_{i-1} \oplus K_i\right)'' \cdot K_i^{odd} \bmod 2^n,$$

де i – номер раунда i = 1..L; K_i, K_i^{odd} – підключі (K_i^{odd} – непарне); $''$ – функція дзеркального відображення.

Розшифрування виконується в оберненому порядку по відношенню до зашифрування.

Процедура зашифрування блокового шифру Caligo здійснюється за таким правилом:

$$Y_i = \left(2^p \cdot K_{3i} \cdot \left(Y_{i-1} \oplus K_{3i+1}\right)\right)'' + K_{3i+2} \bmod 2^n,$$

де K_{3i}, K_{3i+1}, K_{3i+2} – підключі (K_{3i} – непарне); ″ – функція дзеркального відображення; p – допоміжний параметр, який після виконання функції дзеркального відображення, зсуває n-p найбільш значущих бітів на позицію n-p найменш значущих бітів, $0 \le p < n$.

Розшифрування здійснюється таким чином:

$$Y_i = \left(K_{3i}^{-1} \cdot \left(2^p \cdot \left(Y_{i-1} - K_{3i+2} \right) \right)'' \right) \oplus K_{3i+1} \bmod 2^n .$$

Проте, недоліком застосування такого множення, в вище описаних шифрах, як і для множення, яке використано в шифрах [1 – 4, 7 – 9, 11 – 13], що використовує множення за відкритим значенням модуля є його потенційна вразливість до диференційного криптоаналізу із використанням мультиплікативних диференціалів [10]. Оскільки мультиплікативні диференціали розраховуються на підставі відомого значення модуля, який проходить через операції перестановки, додавання за модулем 2^n, операцію виключного-АБО і дозволяє відновити раундові ключі.

В роботах [12, 13] запропонований симетричний блоковий шифр, який оперує з n-бітними блоками даних та ключем, процедура зашифрування яких полягає в виконанні послідовності таких операцій:

$$Y = \left(\left(X \oplus K_1 \right) \cdot K_2 \bmod 2^n \right)^{\leftrightarrow k} \cdot K_3 \bmod 2^n ,$$

де K_1, K_2, K_3 – підключі (K_1, K_3 – непарні); $\leftrightarrow k$ – операція перестановки блоків n-бітного числа по k біт.

Даний блоковий шифр, як і попередні блокові шифри Nimbus і Caligo, використовує операцію множення за модулем 2^n, який вимагає використання непарних множників, а пошук обернених мультиплікативних за модулем 2^n зводиться до вирішення або рівняння (1), або до використання розширеного алгоритма Евкліда.

Найбільш вдалим за своїми конструктивними особливостями шифр, який використовує операцію множення як один із базових криптографічних примітивів є шифр IDEA [4]. В шифрі використовується 128 бітний секретний ключ та 64 бітні блоки даних, які розбиваються на чотири 16 бітних підблоки. Під час розробки шифру було використано три несумісних групи операцій: додавання за модулем 2^{16}, множення за модулем $2^{16}+1$ та операція виключного-АБО. Вони несумісні в тому сенсі, що жодна пара з трьох операцій не задовольняє:

1. Асоціативному закону: $a+(b \oplus c) \ne (a+b) \oplus (a+c)$.
2. Дистрибутивному закону: $a \cdot (b+c) \ne (a \cdot b)+(a \cdot c)$.

Оскільки множення виконується за модулем $m = 2^{16}+1$, який є простим числом, то всі множники будуть взаємно простими з m. В даному шифрі, пошук обернено мультиплікативних за модулем m виконують за допомогою розширеного алгоритма Евкліда. Детальніше з конструктивними особливостями шифру можна ознайомитися в [4].

Отже, використання операції множення в блокових шифрах спричинює таке:

1. Необхідність обчислення оберненого мультиплікативного, з використанням розширеного алгоритму Евкліда або підходу запропонованого в [2], які вимагають достатньо великої кількості операцій.
2. Потенційна вразливість до диференційного криптоаналізу із використанням мультиплікативних диференціалів [10].

Множення за секретним значенням модуля

У роботі [14], проаналізовано можливі варіанти комбінацій секретних модулів та множників у випадку коли раундові функція побудована лише на операції множення. Встановлено, що доцільно використовувати різні значення модулів на кожному раунді, оскільки в такому випадку для двох послідовних раундів буде отримано дві групи несумісних операцій множення за різними модулями, які не задовольнятимуть асоціативному закону:

$$a \cdot (b \cdot c) \ne (a \cdot b) \cdot c.$$

До недоліків розглянутих методів шифрування можна віднести те, що значення модуля на наступному раунді шифрування має бути більшим за попереднє, що зменшує криптографічну стійкість шифру. Окрім того, розрядність модулів повинна бути більшою за розрядність множеного, що ускладнює реалізацію метода та висуваються доволі складні вимоги до формування підключів шифрування, складовими яких є модуль та множник.

Для вирішення проблеми надлишковості вихідного блоку даних та з метою усунення залежностей між модулями пропонується використовувати таке множення за секретним значенням модуля:

$$X \times A \bmod m = \begin{cases} X \cdot f\left(m'\right) \bmod m', & \text{якщо } X < m'; \\ \left(X - m'\right) \cdot f\left(m''\right) \bmod m'' + m', & \text{якщо } X \ge m', \end{cases} \quad (2)$$

де m', m'' – n-бітні модулі, $m' = \overline{2^{n-2}; 3 \cdot 2^{n-2}}$, $m'' = 2n - m'$; $f(m')$, $f(m'')$ – функція, що визначає взаємно прості числа з m' і m'' відповідно.

Оскільки значення модуля залишається невідомим, то це по-перше, дозволяє протидіяти криптоаналізу на основі мультиплікативних диференціалів і по-друге, використання двох послідовних множень за різними модулями дозволяє отримувати групи несумісних операцій.

Якщо замість операції множення «×» в формулу (2) підставити операцію додавання «+», то отримаємо:

$$\left[X + A\right] \bmod m = \begin{cases} \left[X + A'\right] \bmod m', & \text{якщо } X < m'; \\ \left[\left(X - m'\right) + A''\right] \bmod m'' + m', & \text{якщо } X \ge m', \end{cases} \quad (3)$$

де A', A'' – n-бітні доданки, $A' < m'$ і $A'' < m''$.

Як видно з формули (3) доданки A' і A'' є незалежними від значень відповідних модулів m' і m''. Послідовне використання операції додавання згідно

(3) дозволяє отримати дві групи несумісних операцій, які не задовольнятимуть асоціативному закону:

$$a + (b + c) \neq (a + b) + c.$$

Знаходження взаємно простих чисел та обернено мультиплікативних

Авторами роботи [15] пропонується декілька підходів до формування пар взаємно простих чисел. З урахуванням описаних підходів, для того щоб знайти взаємно просте число до деякого випадкового n-бітного модуля $m = \{m_{n-1}, .., m_1, m_0\}$ ($m \geq 5$) достатньо скористатися такою функцією:

$$f(m) = \begin{cases} (m + s_1) >> 1, \text{якщо } m_0 = 1; \\ (m + s_2) >> , \text{якщо } m_0 = m_1 = 0; \\ (m + s_3) >> 1, \text{якщо } m_0 = 0 \text{ і } m_1 = 1, \end{cases} \quad (4)$$

де $s_1 = \pm 1$, $s_2 = \pm 2$, $s_3 = \pm 4$.

Загальна кількість модулів, що може бути використана дорівнює 2^{n-1}, якщо виходити із обмежень, які накладаються на модуль, а саме $m''=2n-m'$ і $2^{n-2} \leq m' \leq 3 \cdot 2^{n-2}$. При цьому, використовуючи формулу (4) для кожного модуля можна сформувати два взаємно простих множника.

Для знаходження значення оберненого мультиплікативного f(m) за модулем m пропонується використовувати таку функцію:

$$f^{-1}(m) =$$

$$= \begin{cases} 2, \text{якщо } s_1 = 1 \text{ і } m_0 = 1; \\ m-2, \text{якщо } s_1 = -1 \text{ і } m_0 = 1; \\ (m+s_2) >> 1, \text{якщо } m_0 = m_1 = 0; \\ (m+s_3) >> 2, \text{якщо } s_3 = 4, m_0 = m_2 = 0, m_1 = 1; \\ (m+f(m)) >> 1, \text{якщо } s_3 = 4, m_0 = 0, m_1 = m_2 = 1; \\ (m-f(m)) >> 1-1, \text{якщо } s_3 = -4, m_0 = 0, m_1 = m_2 = 1; \\ (m+f(m)) >> 1+1, \text{якщо } s_3 = -4, m_1 = 1, m_0 = m_2 = 0. \end{cases}$$

Окрім того, кількість взаємно простих чисел з m можна збільшити використовуючи властивості показника, а саме якщо f(m) і m є взаємно простими числами, то

$$f(m)^{\alpha} \equiv d \bmod m, \quad (5)$$

де α – період генерування взаємно простих чисел, $\alpha \in N$.

Тоді послідовність степенів f(m), $f(m)^2$, $f(m)^3$, …, $f(m)^{\alpha}$ будуть непорівнянні одне з одним за модулем m і утворюватимуть повну зведену систему лишків, якщо f(m) є первісним коренем і неповну зведену систему лишків, в протилежному випадку.

Піднесення до степеня для порівняння (5) зводиться до послідовного множення f(m) на попередній результат $f(m)^{\gamma}$ ($\gamma = 1,...,\alpha$) і відбувається доти, поки $d \neq 1$.

Статистичні дослідження

Для дослідження статистичних властивостей запропонованої операції множення в комбінації з іншими групами операцій, а саме з операцією додавання за модулем 2^{32} та операцією виключного-АБО, обрано метод для проведення статистичних досліджень, який базується на основі наборів тестів NIST STS [16].

Пакет NIST STS складається з 16 статистичних тестів (станом на лютий 2009 NIST рекомендує використовувати 15 тестів), які застосовують для перевірки гіпотези про випадковість двійкової послідовності. Тест направлений на перевірку нульової гіпотези H_0, яка полягає в тому, що послідовність, яка досліджується є випадковою. Також існує альтернативна гіпотеза H_a – досліджувана послідовність не є випадковою. Кожен з тестів підтверджує або спростовує нульову гіпотезу і по результатам всіх тестів приймається загальне рішення про характер появи бітів в заданій послідовності.

Тестування двійкової послідовності M виконується таким чином:

1. Генерується досліджувана послідовність бітів $M = \{S_1, S_2, \ldots, S_m\}$, яка складається з m n-бітних послідовностей S_j, $1 \leq j \leq m$.

2. Для кожної n-бітної послідовності висувається нульова гіпотеза H_0 про те, що послідовність бітів S_j є випадковою.

3. Виконується статистичне дослідження кожної послідовності S_j з отриманням її статистики $\lambda(S_j)$.

4. Для послідовності S_j визначається ймовірність проходження тесту із використанням спеціальної функції $P_j = \delta(\lambda(S_j))$, $P_j \in [0; 1]$.

5. Значення ймовірності P_j порівнюється з рівнем значимості $\upsilon \in [0{,}001; 0{,}01]$. Якщо $P_j \geq \upsilon$, то гіпотеза H_0, приймається. В протилежному випадку гіпотеза відкидається.

6. За сукупністю отриманих значень P_j визначається частка послідовностей S_j, які пройшли гіпотезу H_0 для кожного з q тестів ($1 \leq i \leq q$):

$$r_i = \#\left\{P_j \geq \upsilon \middle| j = \overline{1,m}\right\} \Big/ m .$$

Для кожного отриманого коефіцієнта r_i виконується перевірка послідовності M за таким правилом.

Правило 1. Послідовність M пройшла перевірку для i-го тесту, якщо значення коефіцієнт r_i знаходиться в межах довірчого інтервалу $[r_{min}, r_{max}]$, який визначається за такою формулою:

$$r_{max(min)} = \hat{p} \pm 3\sqrt{\hat{p}(1-\hat{p})/m} ,$$

де $\hat{p} = 1 - \upsilon$.

7. Для кожного і-го тесту на підставі отриманих значень ймовірностей P_j будується гістограма частот F_k потрапляння ймовірності P_j в кожний з $k = \overline{1;10}$ під інтервалів, на які розбитий інтервал [0; 1]. Рівноймовірність розподілу значень ймовірностей P_j, які належать до рівно ймовірного закону розподілу на інтервалі [0; 1], перевіряється за допомогою критерію χ^2. Розрахунок статистики, яка належить до розподілу χ^2 з дев'ятьма степенями свободи виконується за такою формулою:

$$\chi_i^2 = \sum_1^{10} \frac{(F_k - m/10)^2}{m/10}.$$

Правило 2. Послідовність M пройшла тестування для i-го тесту, якщо виконується умова $P\left(\chi_i^2\right) > 0{,}0001$.

8. З урахуванням усіх отриманих результатів приймається рішення, що послідовність M пройшла тестування пакетом NIST STS, якщо значення всіх коефіцієнтів $r_i \in [r_{min}, r_{max}]$ і виконується умова $P\left(\chi_i^2\right) > 0.0001$ для кожного із тестів.

З використанням пакету NIST STS виконано тестування 6 варіантів поєднання трьох груп несумісних операцій для 32-бітних блоків даних: додавання за модулем 2^{32}(A), множення за змінним модулем(M) і побітове додавання за модулем 2(X).Тобто: 1) A→M, 2) X→M, 3) M→A, 4) M→X, 5) A→M→X і 6) X→M→A.

З метою дослідження кожного варіанта поєднання груп несумісних операцій використана циклова функція, одним з параметрів якої є кількість циклів перетворення L, що змінюється для кожної досліджуваної послідовності M. Так, для першого варіанту використано таку циклову функцію:

$$Y_{(i)(j)} = \left(Y_{(i-1)(j)} + A_{(i)(j)}\right) \cdot f\left(B_{(i)(j)}\right) \bmod B_{(i)(j)},$$

де $i = \overline{1;L}$, $L = \overline{1;32}$; j – номер блока даних, який генерується; $Y_{(0)(0)} = 0$ – початковий стан, $Y_{(0)(j)} = Y_{(L)(j-1)}$.

Коефіцієнти $A_{(i)(j)}$ і $B_{(i)(j)}$ генеруються на двох незалежних генераторах псевдовипадкових чисел. Для варіантів 5 і 6 використано три генератора псевдовипадкових чисел.

При цьому, кожна із сформованих послідовностей M досліджена із 10-ма змінними початковими заповненнями генераторів псевдовипадкових чисел. Загальна кількість досліджених послідовностей M для одного варіанта поєднання груп несумісних операцій рівна 32·10 = 320.

Для виконання тестування обрано такі параметри:

1. Для кожного з варіантів поєднання груп несумісних операцій довжина досліджуваної послідовності рівна $n = 10^6$ біт.

2. Кількість досліджуваних послідовностей m = 100. Загальний об'єм вибірки рівний $N = 10^6 \cdot 10^2 = 10^8$.

3. Рівень значимості $\upsilon = 0{,}01$.

4. Кількість тестів q = 188. Результати спектрального тесту на основі дискретного перетворення Фур'є не враховувалися.

Отже, для заданого коефіцієнта значимості $\upsilon = 0{,}01$ може бути відкинута лише одна досліджувана послідовність з m = 100. Оскільки, це є ідеальний випадок, то на практиці використовуються довірчий інтервал $[r_{min}, r_{max}]$ для коефіцієнтів r_i, $r_{min} = 0{,}96015$, а $r_{max} = 1{,}0$.

На підставі вище наведених правил та обраних параметрів отримано такі результати (табл. 1). В табл. 1 занесені результати проходження послідовностей M наборі тестів NIST STS для 10 різних заповнень генераторів псевдовипадкових чисел.

Таблиця 1

Результати тестування поєднання груп несумісних операцій

L	Комбінація операцій					
	AM	XM	MA	MX	AMX	XMA
1	0	0	2	0	1	3
2	0	0	2	0	2	2
3	**6**	2	2	3	4	4
4	**7**	2	4	**7**	3	**6**
5	3	3	4	4	3	2
6	3	3	4	3	1	**5**
7	**6**	**5**	1	2	3	4
8	**5**	4	4	3	**7**	3
9	**8**	2	3	4	3	3
10	3	**5**	3	2	**5**	4
11	1	3	4	**6**	**8**	3
12	**5**	3	3	1	3	**5**
13	3	3	1	3	4	**6**
14	4	3	1	**6**	4	4
15	3	4	3	4	4	2
16	4	**5**	2	1	1	4
17	4	2	3	3	4	**5**
18	**5**	4	**6**	2	**5**	**7**
19	4	**6**	2	3	**7**	2
20	4	**5**	2	4	4	**5**
21	4	3	3	2	**5**	4
22	**5**	4	1	3	3	3
23	2	**5**	1	1	2	**6**
24	3	4	2	4	4	1
25	**5**	1	**6**	4	2	3
26	4	0	**5**	4	4	3
27	**5**	**6**	4	4	2	2
28	**6**	**5**	4	3	1	2
29	2	**6**	2	4	3	4
30	**5**	4	**5**	3	2	**6**
31	**6**	0	1	3	0	**5**
32	3	2	2	3	2	3

Так, поєднання групи операцій A→M з використанням 9 циклів перетворення та групи операцій A→M→X з 11 циклами перетворення дають найкращі статистичні характеристики для різних заповнень генераторів псевдовипадкових чисел, що свідчить про їхню криптографічну стійкість до статистичного криптоаналізу. Окрім того, непогані статистичні характеристики дають й інші розглянуті комбінації арифметичних операцій для 3-4 циклів перетворення.

Також, було проведено статистичне дослідження поєднання найкращих комбінацій арифметичних операцій для 64-бітних блоків даних. Виявилося, що отримані результати для 64-бітних блоків даних є гіршими ніж для 32-бітних. Це пояснюється тим, що для виконання 64-бітних операцій з 64-бітними блоками даних використовувалися інші генератори псевдовипадкових чисел і початкові заповнення для формування 64-бітних коефіцієнтів $A_{(i)(j)}$ і $B_{(i)(j)}$.

Висновки

В роботі виконано аналіз структури існуючих симетричних блокових шифрів, який дозволив виділити існуючі проблеми, що стоять перед розробниками шифрів під час використання операції множення за модулем та запропонувати шляхи їх вирішення.

Недоліком застосування множення за відкритим модулем є його потенційна вразливість до диференційного криптоаналізу із використанням мультиплікативних диференціалів. Оскільки мультиплікативні диференціали розраховуються на підставі відомого значення модуля, який проходить через різні групи операції.

Окрім того, використання операції множення в блокових шифрах спричинює необхідність обчислення оберненого мультиплікативного, з використанням розширеного алгоритму Евкліда або підходу запропонованого А.В. Мачадо, які вимагають достатньо великої кількості операцій.

Запропоновано операцію множення за відкритим модулем замінити на операцію множення за секретним модулем, що в свою чергу дозволяє збільшити кількість підключів, які можуть бути використані під час шифрування та протидіяти диференційному криптоаналізу на основі мультиплікативних диференціалів.

Запропонована функція для знаходження взаємно простого числа для заданого модуля m і відповідна функція для знаходження оберненого мультиплікативного для заданого модуля є більш швидкою ніж розширений алгоритм Евкліда та розглянутий метод А.В. Мачадо.

Застосування вище розглянутого модульного множення та функції знаходження взаємно простих чисел дозволяють будувати процедури розширення ключів «на льоту».

Список літератури

1. *MARS – a candidate cipher for AES [Електронний ресурс] / C. Burwick and others // IBM Corporation, 1999. – P. 63. – Режим доступу до ресурсу: http://www.research.ibm.com/security/mars.pdf*
2. *Machado A.W. The Nimbus Cipher [Електронний ресурс] / A.W. Machado // New European Schemes for Signatures, Integrity, and Encryption, 2000. – P. 7. – Режим доступу до ресурсу: http://www.cosic.esat.kuleuven.be/nessie/workshop/submissions/nimbus.zip.*
3. *Machado A.W. Caligo, an Extensible Block Cipher and CHash, a Caligo Based Hash [Електронний ресурс] / A.W. Machado // Cryptographic Hash Algorithm Competition, 2006. – P. 11. – Режим доступу до ресурсу: csrc.nist.gov/groups/ST/hash/documents/MACHADO_caligo.pdf.*
4. *Lai X. A Proposal for a New Block Encryption Standard / Xuejia Lai, James L. Massey // EUROCRYPT, 1990. – P. 389-404.*
5. *Daemen J. Block ciphers based on modular arithmetic / J. Daemen, R. Govaerts, J. Vandewalle // In Proc. of the 3rd symp. on State and Progress of Research in Cryptography, W. Wolfowicz (ed.), Fondazione Ugo Bordoni, 1993. – P. 80-89.*
6. *Schmidt D. ABC - A Block Cipher [Електронний ресурс] / D. Schmidt // Cryptology ePrint Archive – 2002. – P. 50. – Режим доступу до ресурсу: http://eprint.iacr.org/2002/062.*
7. *Xenon 1.0: Architecture and Specification [Електронний ресурс] / Chang-Hyi Lee and others // Crypto-Laboratory of SoftForum. – 2000. – P. 41. – Режим доступу до ресурсу: http://web.archive.org/web/20031024050027/http://www.softforum.com/english/download/Xenon_V1.0.pdf.*
8. *The RC6 Block Cipher [Електронний ресурс] / R.L. Rivest, M.J.B. Robshaw, R. Sidney, Y.L. Yin // New European Schemes for Signatures, Integrity, and Encryption, 2000. – P. 16. – Режим доступу до ресурсу: https://www.cosic.esat.kuleuven.be/nessie/workshop/submissions/rc6.zip.*
9. *M'Raihi D. xmx: A Firmware-Oriented Block Cipher Based on Modular Multiplications / David M'Raihi and others // 4th International Workshop on Fast Software Encryption (FSE '97). – Haifa: Springer-Verlag. – P. 166-171.*
10. *Multiplicative differentials / N. Borisov, M. Chew, R. Johnson, D. Wagner // Fast Software Encryption: 9th Int. Workshop on table of contents. – 2002. – Vol. 2365. – P. 17-33.*
11. *Screamer B. Microsoft's digital rights management scheme – technical details [Електронний ресурс] / B. Screamer. – 2001. – Режим доступу: http://cryptome.org/ms-drm.htm.*
12. *Сокирук В.В. Побудова статистично безпечного БСШ на основі арифметичних операцій за модулем / В.В. Сокирук, В.А. Лужецький // Інформаційні технології та комп'ютерна інженерія. – В.: ВНТУ, 2006. – № 1. – С. 158-163.*
13. *Сокирук В.В. Програмна реалізація блокового симетричного шифру на основі арифметичних операцій за модулем 2^n / В.В. Сокирук, В.А. Лужецький // Вісник ВПІ. – В.: ВНТУ, 2007. – № 2. – С. 80-85.*
14. *Лужецький В.А. Блокові шифри для режиму роботи ECB / В.А. Лужецький, О.В. Дмитришин // Інформаційні технології та комп'ютерна інженерія. – В.: ВНТУ, 2008. – № 1. – С. 154-158.*
15. *Лужецький В.А. Процедури розгортання ключів для блокових шифрів на основі арифметичних операцій за модулем / В.А. Лужецький, О.В. Дмитришин // Інформаційні технології та комп'ютерна інженерія. – В.: ВНТУ, 2009. – № 2. – С. 69-74.*
16. *A Statistical Test Suite for Random and Pseudorandom Number Generator for Cryptographic Application. NIST Special Publication 800-22, August 2008 / Andrew Rukhin, and others. – P. 131.*

Надійшла до редколегії 30.04.2010

Рецензент: д-р техн..наук, проф. В.М. Рудницький, Черкаський державний технологічний університет, Черкаси.

ИСПОЛЬЗОВАНИЕ ОПЕРАЦИИ УМНОЖЕНИЯ ПО МОДУЛЮ В СИММЕТРИЧНЫХ БЛОЧНЫХ ШИФРАХ

В.А. Лужецкий, О.В. Дмитришин

Рассмотрен один из методов использования операции умножения по модулю в блочных шифрах. Предложены методы генерирования взаимно простых чисел и поиска обратно мультипликативных.

Ключевые слова: *блочный шифр, умножение по модулю, взаимно простые числа, обратно мультипликативные.*

USING OF THE MODULAR MULTIPLICATION IN THE SYMMETRIC BLOCK CIPHERS

V.A. Luzhetsky, O.V. Dmytryshyn

One of the methods of modular multiplication in block ciphers is considered. Methods of co-prime numbers generating and multiplicative inverse search are proposed.

Keywords: *sectional code, increase on the module, mutually prime numbers, back multiplicative.*