НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
Навчально-науковий фізико-технічний інститут
Кафедра математичних методів захисту інформації

«На правах рукопису»
УДК 004.056.55:512.6+519.24

«До захисту допущено»
В.о. завідувача кафедри
_____ Сергій ЯКОВЛЄВ
«__» _____ 2022 р.

Магістерська дисертація

на здобуття ступеня магістра
за освітньо-професійною програмою
«Математичні методи криптографічного захисту інформації»

зі спеціальності: 113 Прикладна математика
на тему: **«Алгоритм оцінювання стійкості небінарних SP-мереж до узагальненого лінійного криптоаналізу»**

Виконав:
студент II курсу, групи ФІ-12мп
Тафтай Анастасія Олексіївна _____

Керівник:
доцент кафедри ММЗІ, к.т.н.
Яковлєв Сергій Володимирович _____

Рецензент:
доцент кафедри ІБ, к.т.н.
Стьопочкіна Ірина Валеріївна _____

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.

Студент _____

Київ — 2022

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
Навчально-науковий фізико-технічний інститут
Кафедра математичних методів захисту інформації

Рівень вищої освіти — другий (магістерський)
Спеціальність — 113 Прикладна математика,
ОПП «Математичні методи криптографічного захисту інформації»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ Сергій ЯКОВЛЄВ
«__» _____ 2022 р.

ЗАВДАННЯ
на магістерську дисертацію

Студент: Тафтай Анастасія Олексіївна

1. Тема роботи: «*Алгоритм оцінювання стійкості небінарних SP-мереж до узагальненого лінійного криптоаналізу*», науковий керівник дисертації: доцент кафедри ММЗІ, к.т.н. Яковлєв Сергій Володимирович,

затверджені наказом по університету №__ від «__» _____ 2022 р.

2. Термін подання студентом роботи: «__» _____ 2022 р.

3. Об'єкт дослідження: інформаційні процеси в системах криптографічного захисту інформації

4. Предмет дослідження: методи оцінювання стійкості SP-мереж до узагальненого лінійного криптоаналізу

5. Перелік завдань:

1) провести огляд опублікованих джерел за тематикою дослідження;

2) дослідження аналітичних оцінок та алгоритмів оцінювання стійкості SP-мереж до лінійного криптоаналізу;

3) імплементація алгоритму побудови оцінок для обчислення нижньої границі максимального лінійного потенціалу;

4) уточнення оцінок стійкості шифру Midori64 до узагальненого лінійного криптоаналізу за допомогою імплементованих алгоритмів.

6. Орієнтовний перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): Презентація доповіді

7. Орієнтовний перелік публікацій: планується доповідь на всеукраїнській конференції

8. Дата видачі завдання: 10 вересня 2021 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання	Примітка
1	Узгодження теми роботи із науковим керівником	01-15 вересня 2022 р.	Виконано
2	Огляд опублікованих джерел за тематикою дослідження	вересень-жовтень 2022 р.	Виконано
3	Дослідження та імплементація алгоритму для оцінки стійкості до лінійного криптоаналізу	жовтень-листопад 2022 р.	Виконано
4	Отримання результатів та їх аналіз	листопад-грудень 2022 р.	Виконано

Студент _____ Анастасія ТАФТАЙ

Керівник _____ Сергій ЯКОВЛЄВ

РЕФЕРАТ

Кваліфікаційна робота містить 38 сторінок, 2 рисунки, 4 таблиці, 14 джерел.

Метою магістерскої роботи є вдосконалення методів оцінювання стійкості небінарних SP-мереж до лінійного криптоаналізу. Об'єктом дослідження є інформаційні процеси в системах криптографічного захисту інформації.

Предметом дослідження є методи оцінювання стійкості SP-мереж до лінійного криптоаналізу.

В роботі роботи проведено огляд опублікованих джерел за тематикою дослідження, зокрема дослідження аналітичних оцінок та алгоритмів оцінювання стійкості SP-мереж до лінійного криптоаналізу.

Також узагальнено метод Келіхера для побудови оцінок максимального значення узагальненого середнього лінійного потенціалу для небінарних SP-мереж, орієнтованих на модульне додавання. Вперше одержано оцінки стійкості небінарних версій шифру Midori64 до узагальненого лінійного криптоаналізу.

СИМЕТРИЧНА КРИПТОГРАФІЯ, ЛІНІЙНИЙ КРИПТОАНАЛІЗ, УЗАГАЛЬНЕНИЙ ЛІНІЙНИЙ КРИПТОАНАЛІЗ, НЕБІНАРНІ ШИФРИ, ШИФР MIDORI

ABSTRACT

Qualification work contains: 38 pages, 2 figures, 4 tables, 14 sources.

The purpose of this thesis is the improvement of methods of security evaluations of SP-networks against linear cryptanalysis. The object of research is information processes in systems of cryptographic protection of information. The subject of research is methods for assessing security evaluations of SP-networks to linear cryptanalysis. par

In the course of writing the qualification work, a review of published sources on the research topic of analytical estimations and algorithms of estimation of resistance of SP-networks to linear cryptanalysis was conducted.

Also Keliher's technique for computing bounds on maximum expected (generalized) linear potential for non-binary SP-networks based on modular addition was generalized. The first security evaluation of non-binary versions of Midori64 against generalized linear cryptanalysis was presented.

SYMMETRIC CRYPTOGRAPHY, LINEAR CRYPTANALYSIS, GENERALIZED LINEAR CRYPTANALYSIS, NON-BINARY CIPHERS, MIDORI CIPHER

ЗМІСТ

ВСТУП

Актуальність дослідження.

Стійкість до лінійного криптоаналізу є одною з важливих характеристик при розробці, використанні і аналізі сучасних блокових шифрів. Вона визначається насамперед через значення такої характеристики шифру як максимальний середній лінійний потенціал. Точні значення такої характеристики важко порахувати, тому для неї шукаються деякі обмеження, що допомагають сформувати уявлення про стійкість криптосистеми до лінійного криптоаналізу. Звичайний лінійний криптоаналіз застосовний лише для небінарних шифрів. Тож узагальнення його дозволить сформувати оцінки стійкості і для шифрів з цієї категорії.

Метою дослідження є вдосконалення методів оцінювання стійкості небінарних SP-мереж до лінійного криптоаналізу. Для досягнення мети необхідно виконати такі завдання:

1) провести огляд опублікованих джерел за тематикою дослідження;

2) дослідження способів узагальнення лінійного криптоаналізу для небінарних шифрів;

3) дослідження оцінок та алгоритмів оцінювання стійкості SP-мереж до лінійного криптоаналізу;

4) реалізація алгоритму побудови оцінок для обчислення нижньої границі максимального лінійного потенціалу;

5) уточнення оцінок стійкості шифру Midori64 до узагальненого лінійного криптоаналізу за допомогою реалізованого алгоритму.

Об'єктом дослідження є інформаційні процеси в системах криптографічного захисту інформації.

Предметом дослідження є методи оцінювання стійкості SP-мереж до узагальненого лінійного криптоаналізу.

При розв'язанні поставлених завдань використовувались такі

методи дослідження: теорії імовірностей, алгебраїчні методи, методи комп'ютерного та математичного моделювання.

Наукова новизна. Вперше застосовано алгоритм оцінювання нижньої границі максимального середнього лінійного потенціалу до шифру Midori. Одержано значення нових оцінок стійкості шифру Midori до лінійного криптоаналізу.

Практичне значення. Результати даної роботи можуть бути використані для уточнення рівня захищеності систем криптографічного захисту на основі шифру Midori.

1 ТЕОРЕТИЧНІ ВІДОМОСТІ ТА ОГЛЯД ЛІТЕРАТУРИ

У розділі було введено деякі поняття для оцінок стійкості шифрів до лінійного криптоаналізу, таких як лінійний потенціал та максимальний середній лінійний потенціал. Досліджено методи узагальнення лінійного криптоаналізу на небінарні шифри. Також наведено опис структури блокових шифрів і їх класифікацію, а також специфікації блокового небінарного шифру Midori.

1.1 Поняття Блокового Шифру

Як очевидно з назви, блокові шифри оперують над "блоками"даних фіксованої довжини, зазвичай це 64 чи 128 бітів, які трансформуються у блоки такої ж довжини в результаті шифрування.

Означення 1.1. Нехай M – множина відкритих текстів, C – множина шифротекстів, K – множина ключів. *Шифруючим перетворенням* називається функція виду

$$f : M \times K \to C$$

така що для кожного фіксованого значення $k \in K$ перетворення $f_k(x) = f(x, k)$ є бієктивним.

Зазвичай множини відкритого тексту і шифротекстів збігаються і є множинами бітових чисел довжини блоку шифру.

Означення 1.2. Нехай M – множина відкритих текстів, C – множина шифротекстів, K – множина ключів. *Ітеративним r-раундовим блоковим шифром* називається перетворення $E : M \times K^r \to C$ що є композицією r шифруючих перетворень

$$f : M \times K \to C$$

така що для кожного фіксованого значення $k \in K$ перетворення $f_k(x) = f(x, k)$ є бієктивним.

Блокові ітеративні шифри були запропоновані та аналізовані Клодом Шеноном у 1949 як спосіб збільшення стійкості комбінуючи прості операції як розсіювання (рівноймовірність всіх статистик шифротексту) та перемішування (кожен біт відкритого тексту й секретного ключа повинен впливати на якомога більше бітів шифротексту). Існує два способи імплементації цих принципів, які виділяють блокові ітеративні виділяють шифри на дві категорії: мережа Фейстеля та SP-мережа.

Означення 1.3. Нехай $x \in F_q^n$, $n = mt$. *SP-мережею* називається ітеративний блоковий шифр, який складається з кількох раундів, кожен з яких визначається як

$$F_k(x) = L(S(K(x, k)))$$

де L – лінійне відображення, $S(x) = (s_1(x_1), s_2(x_2), \ldots, s_m(x_m))$ – нелінійний шар, який складається з бієктивних перетворень $s_i : F_q^t \to F_q^t$, які називаються S-блоками, K – функція замішування з ключем (Рисунок 1.1).

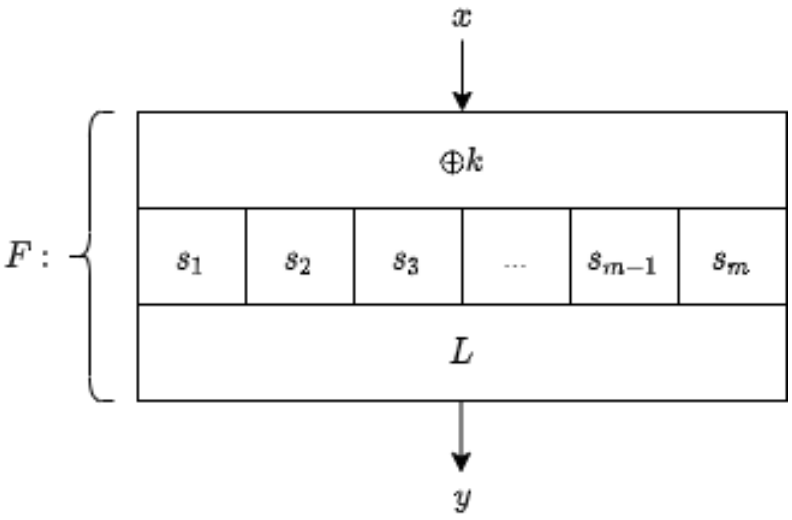

Рисунок 1.1 – Раунд SP-мережі

У більшості випадків K є функцією побітового додавання, а L - матричне перетворення.

Також виділяють категорію малоресурсних блокових шифрів, як очевидно з назви, тих які потребують невелику кількість ресурсів й можуть виконуватись в обмежених середовищах де виконання звичайних шифрів є неефективним або неможливим.

1.2 Специфікації шифру Midori

Шифр Midori [2] є сімєю з двох малоресурсних блокових шифрів: Midori64 і Midori128. Кожен з них приймає ключ довжиною 128 бітів, і мають різну довжину блоку – 64 і 128 бітів відповідно.

Таблиця 1.1 – Характеристики шифру Midori

	розмір блоку(n)	розмір ключа	розмір комірки (m)	кількість раундів
Midori64	64	128	4	16
Midori128	128	128	8	20

Шифр за структурою є SP-мережею, й використовує масив розміру 4x4 який називається станом:

$$S = \begin{bmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \end{bmatrix}$$

де розмір кожної клітинки дорівнює m (4 чи 8) бітів, відповідно $s_i \in \{0,1\}^m$

На і-тому раунді стан є S_i, відповідно якщо відкритий текст дорівнює P то $S_0 = P$.

Як будь-яка SP-мережа, Midori складається з нелінійного та лінійного шару. Нелінійний шар – бієктивні S-блоки $\{0,1\}^4 \to \{0,1\}^4$. Мідорі має 2 варіанти S-блоків: Sb_0, Sb_1 які використовуються в Midori64 та Midori128 відповідно. S-блоки застосовуються до кожної 4 чи 8 бітної клітинки матриці стану паралельно.

Таблиця 1.2 – S-блоки шифру Midori

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$Sb\_0[x]$	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6
$Sb\_1[x]$	1	0	5	3	e	2	f	7	d	a	9	b	c	8	4	6

Лінійний шар шифру утворюють функції ShuffleCell та MixColumn: $\{0,1\}^n \to \{0,1\}^n$ Функція ShuffleCell перемішує значення клітинок матриці стану наступним чином:

$$(s_0, s_1, ..., s_{15}) \leftarrow (s_0, s_{10}, s_5, s_{15}, s_{14}, s_4, s_{11}, s_1, s_9, s_3, s_{12}, s_6, s_7, s_{13}, s_2, s_8)$$

MixColumn своєю чергою є застосуванням перетворення M до кожного 4m-бітної колонки матриці стану

$$^t(s_i, s_{i+1}, s_{i+2}, s_{i_3}) \leftarrow M^t(s_i, s_{i+1}, s_{i+2}, s_{i_3}),\ i = 0,4,8,12$$

Шифр пропонує 3 варіанти перетворень M: M_A, M_B, M_C, перші ша з яких є інволютивною MDS, друга – не інволютивною MDS, третя - інволютивна майже MDS.

$$M_A = \begin{bmatrix} 1 & 2 & 6 & 4 \\ 2 & 1 & 4 & 6 \\ 6 & 4 & 1 & 2 \\ 4 & 6 & 2 & 1 \end{bmatrix} M_B = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} M_C = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

Додавання з раундовим ключем реалізовано через операцію побітового додавання.

Алгоритм 1.1.
procedure MIDORICODE(s_i)
    S = KeyAdd(M, WK)
    for i = 0, ..., R-2 **do**
        S = SubCell(S)
        S = ShuffleCell(S)
        S = MixColumn(S)
        S = KeyAdd(S, RK_i)
    end for
    S = SubCell(S)
    C = KeyAdd(S)
end procedure

1.3 Поняття лінійного криптоаналізу

Лінійний криптоаналіз вивчає статистичні залежності між бітами відкритого тексту, шифротексту та ключами, якими були вони зашифровані. Ці залежності використовуються для припускання значень бітів ключа, коли у зловмисника наявно багато наборів відкритих текстів з відповідними їм шифротекстами, тобто при атаці з використанням відкритого тексту. Ідея атаки полягає в побудові лінійної апроксимації

шифру, тобто знаходження "ефективного"виразу для даного шифру

$$\alpha dot x = \beta dot y$$

де x, y - відкритий тест та шифротекст, а α, β - їхні маски відповідно. Нехай

$$\Pr(\alpha \cdot x = \beta \cdot y) = p$$

Потрібно, щоб ймовірність p була меншою за $1/2$.[9]

Вперше лінійний криптоаналіз було запропоновано Мацуї для побудови атаки на FEAL та DES[3, 4], в основі якої лежить пошук співвідношення між відкритими текстами, шифротекстами й ключами, які можуть бути використані для пошуку бітів ключа через розв'язування систем рівнянь чи побудови атаки розпізнавання на ключі останніх раундів ітеративного блокового шифру.

Означення 1.4. *Лінійним потенціалом* називається величина

$$LP^f(\alpha,\beta) = (2 \cdot \Pr(\alpha \cdot x = \beta \cdot f(x)) - 1)^2$$

де α, $\beta, x \in V_n$, $f : V_n \to V_n$ - деяка булева функція.

для бінарних шифрів цей вираз можна переписати як:

$$LP^f(\alpha,\beta) = (\frac{1}{2^n}\sum_x (-1)^{\alpha x \oplus \beta f(x)})^2$$

Очевидно, що для наведеної величини виконуються наступна рівність:

$$\sum_{u \in \{0,1\}^d} LP(\alpha, u) = \sum_{b \in 0,1^d} LP(b, \beta) = 1$$

Означення 1.5. *Середнім (лінійним) потенціалом* є усереднення лінійних потенціалів за значеннями ключів для параметризованих

ключами функцій $f_k : V_n \times K \to V_n$:

$$ELP^{f_k}(\alpha,\beta) = \frac{1}{|K|} \sum_{k \in K} LP^{f_k}(\alpha,\beta)$$

Стійкість до лінійного криптоаналізу визначається максимальним потенціалом шифруючого перетворення $MELP(f_k) = \max_{\alpha,\ \beta \neq 0} ELP^{f_k}(\alpha,\beta)$ і є обернено пропорційна його значенню: верхні межі цього значення задають нижні границі складності для лінійного криптоаналізу.

1.4 Узагальнений лінійний криптоаналіз

Формальна теорія криптоаналізу, розвинута лише для блокових шифрів які оперують бітовими строками як відкритим текстом, тобто марківських відносно побітового додавання. Лише для таких шифрів існують певні аналітичні оцінки стійкості. Відповідно, враховуючи існування небінарних шифрів, цей метод аналізу потребує узагальнення на такі типи шифрів.

Існує кілька напрямків можливих узагальнень лінійного криптоаналізу. У [5] пропонуєтеся замість лінійних апроксимацій використовувати довільні збалансовані відображення. У [6] розглядають білінійні апроксимації блокових шифрів. Існують також складніші модифікації – аналіз нульових кореляцій, диференційних атак тощо. Для шифрів з не побітовим додаванням у ключовому суматорі у [7] було запропоновано використовувати спеціальні параметри S-блоків, що дозволило побудувати обґрунтовані оцінки практичної стійкості до класичного лінійного криптоаналізу. Зовсім інший підхід пропонують Байнере, Стерн, Водено [8] для шифрів, які побудовані з орієнтацією на деяку абелеву групу з операцією додавання, яка не є побітовим додаванням, шляхом узагальнення поняття лінійної апроксимації, деталі

якого наведені нижче.

Означення 1.6. *Характером* групи G називається довільний гомоморфізм $\chi : G \to \mathbb{C}^*$, де $\mathbb{C}^*$ – мультиплікативна група ненульових комплексних чисел.

У просторі $0{,}1^n$ задамо певну операцію $+$, яка визначає структуру абелевої групи $G = \langle 0{,}1^n, + \rangle$. визначену групу характерів $\hat{G}$, дуальну до G, яка може бути пронумерована двійковими векторами з $0{,}1^n$.
Розглянемо випадок коли $G = \{0,1\}^k$, $\chi_u(a) = (-1)^{u \cdot a}$ для всіх $a, u \in G$ і де $\cdot$ означає скалярний добуток к групі G. Відображення $u \longmapsto \chi_u$ є ізоморфізмом між групами G та $\hat{G}$. Отже, коли $G = \{0,1\}^k$ будь-який характер цієї групи боже бути вираженим як $\chi(a) = (-1)^{u \cdot a}$ для деякого $u \in G$ В звичайному лінійному криптоаналізі u називається маскою і існує бієкція між масками і характерами. Отже, виглядає ґрунтовним узагальнювати лінійний криптоаналіз на будь-яку скінчену абелеву групу використовуючи характери замість масок.

В попередньому розділі було наведене визначення лінійного потенціалу, яке можна переписати як

$$LP_D(u) = (2 \Pr_{X \in D\{0,1\}^n}[u \cdot X = 0] - 1)^2 = (E_{X \in D\{0,1\}^n}((-1)^{u \cdot X}))^2$$

де u – маска, n – розмір блоку, D – розподіл

Означення 1.7. *Узагальнений лінійний потенціал* характеру $\chi : G \to \mathbb{C}^\times$, для деякого розподілу D над G визначається як

$$LP_D(\chi) = |\sum_{a \in G} \chi(a) \Pr_D(a)|^2 = (E_{A \in DG}(\chi(A))^2$$

Видно, що χ є квадратною магнітудою перетворення Фурє ймовірнісного розподілу. Очевидно, що для будь-якого u виконується рівність $LP_D(u) = LP_D(\chi_u)$, а отже цей лінійний потенціал є дійсно узагальненим. У випадку коли $G = \mathbb{Z}_{2^n}$, $n > 1$ маємо характери

наступного виду: $\chi(x) = e^{\frac{2i\pi}{n}x}$

Отже, узагальнений лінійний потенціал набуває виду:

$$LP^f(\alpha,\beta) = (\frac{1}{2^n}|\sum_{x\in V_n}\alpha(x)\cdot\overline{\beta(f(x))}|)^2 = (\frac{1}{2^n}|\sum_{x\in V_n}e^{\frac{2\pi i}{2^n}(\alpha x-\beta(f(x)))}|)^2$$

Узагальнені лінійні потенціали мають такі ж самі властивості як і звичайні, що дозволяє переформувати алгоритми атаки і означення параметрів стійкості зі звичайного криптоаналізу:

$$MELP(E) = \max_{\alpha,\ \beta\in\hat{G},\beta\neq 0} ELP^E(\alpha,\ \beta)$$

1.5 Оцінювання стійкості шифрів до лінійного криптоаналізу

Наведемо деякі поняття лінійного криптоаналізу.

Означення 1.8. *Лінійною характеристикою* для раундів 1...T називається (T+1)-кортеж з N-бітових масок $\Omega = \langle a^1, a^2, ..., a^T, a^{T+1}\rangle$, де a^t, a^{t+1} – вхідні і вихідні маски раунду t $(1 \leqslant t \leqslant T)$.

$$LP^f(\alpha,\beta) = (2\cdot\Pr(\alpha\cdot x = \beta\cdot f(x)) - 1)^2$$

де $\alpha,\ \beta, x \in \{1,0\}^n$, $f : \{1,0\}^n \to \{1,0\}^n$ - деяка булева функція.

Означення 1.9. *Середнім лінійним потенціалом характеристики* Ω називається величина

$$ELCP^{[1...T]}(\Omega) = \prod_{t=1}^{T}(a^t,\ a^{t+1})$$

Означення 1.10. Якщо $T \geqslant 2, a,b \in \{1,0\}^n$, то відповідною *лінійною оболонкою* $ALH(a,\ b)$ є множина всіх лінійних характеристик для раундів 1...T такі що a є першою вхідною маскою , a b – останньою, вихідною, тобто всі лінійні характеристики виду $\Omega = \langle a, a^1, ..., a^T, b\rangle$

Теорема 1.1. *Нехай* $a,\ b \in \{1,0\}^n$. *Тоді*

$$ELP^{[1...T]}(a,\ b) = \sum_{\Omega \in ALH(a,\ b)} ELCP^{[1...T]}(\Omega)$$

Повертаючись до SP-мереж, виразимо середній лінійний потенціал в термінах її компонентів, таких як S-блоки.

Лема 1.1. *Нехай* L *– шар лінійного перетворення мережі, представлений у вигляді матриці* $n \times n$; x, y *– вхід та вихід для лінійної трансформації (транспоновані вектори), відповідно* $y = Lx$. *Тоді якщо* $a \in \{1,0\}^n$ *– маска вля входу перетворення* L, *тоді існує єдина така маска* $b \in \{1,0\}^n$ *що для всіх* $x \in \{1,0\}^n$: $a \cdot x = b \cdot (Lx)$. *Відношення між* a *та* b *таке, що* $a = L^T b$, *де* L^T *– транспонована матриця* L.

З леми випливає, що якщо a^t та a^{t+1} вхідні та вихідні маски для раунду t, то результуючими масками для нелінійного шару раунду t є a^t, $b^t = L^T a^{t+1}$. надалі a^t та b^t можуть бути легко розділені на маски для кожного S-блоку шифру. Пронумеруємо s-блоки як $S_1^t, S_2^t, ..., S_M^t$ та позначимо відповідні їм вхідні і вихідні маски як $a_m^t, b_m^t (1 \leqslant m \leqslant M)$ тоді з теореми Мацуї [**?**]

$$ELP^t(a^t, b^t) = \prod_{m=1}^{M} LP^{S_m^t}(a_m^t,\ b_m^t)$$

Означення 1.11. Нехай Ω є T-раундовою лінійною характеристикою для раундів 1...T. Тоді Ω називається *консистентною, якщо для кожного S-блоку* раундів 1...T вхідні та вихідні маски визначаються нею для цього S-блоку як або обидві нульові або обидні не нульові.

Означення 1.12. Маючи консистентну лінійну характеристику, кожен S-блок для якого результуючі вхідні і вихідні маски ненульові називається *активним*

Надалі будемо говорити тільки про консистентні характеристики.

Означення 1.13. Нехай $v \in \{1,0\}^n$ – вхідна чи вихідна маска для

нелінійного шару раунду t. Тоді активні S-блоки на раунді t можна отримати з цієї маски (без знання відповідної вихідної чи вхідної маски). Позначимо γ_v M-бітний вектор який кодує шаблон активних S-блоків: $\gamma_v = \gamma_1\gamma_2...\gamma_M$, де $\gamma_i = 1$ якщо i-тий S-блок є активним і 0 якщо ні.

Означення 1.14. Нехай $\gamma, \hat{\gamma} \in \{0,1\}^M$. Тоді

$$W_l[\gamma, \hat{\gamma}] = \{y \in \{1,0\}^n : \gamma_x = \gamma, \gamma_y = \hat{\gamma}, x = L^T y\}$$

Неформально, це значення означає кількість способів лінійної трансформації для зв'язку шаблону активних S-блоків в один раунд (γ) з шаблоном активних S-блоків в наступному раундові ($\hat{\gamma}$).

Означення 1.15. Лінійним індексом розгалуження SP-мережі називається мінімальна кількість активних S-блоків за 2 послідовних раунди для будь-якої не нульової характеристики:

$$B_l = \min\{wt(\gamma_x) + wt(\gamma_y) : y \in \{1,0\}^n\ 0,\ x = L^T y\}$$

Очевидно, що $2 \leqslant B_l \leqslant (M+1)$

Протягом останніх років, було опубліковано ряд досліджень стосовно стійкості SP-мереж до лінійного криптоаналізу.

Стійкість до лінійного криптоаналізу визначається максимальним лінійним потенціалом шифруючого перетворення MELP і є обернено пропорційна його значенню: верхні межі цього значення задають нижні границі складності для лінійного криптоаналізу. Значення MELP визначається наступним чином:

$$MELP(E) = \max_{a,\ b \in 1,0^n, b \neq 0} ELP^{[1...T]}(a, b)$$

де T – кількість основних раундів ($T = R - 1$ чи $T = R - 1$, R– кількість раундів шифру). В загальному, для $T \geqslant 2$ дуже важко обчислити точне

значення MELP, тому мають місце визначення границь цього значення.

Лема 1.2. *Нехай*

$$QA_{S_i} = max_a \sum_b (ord(b) - 1) \cdot (LP_i^S(a,b))_l^B$$
$$QB_{S_i} = max_b \sum_a (ord(a) - 1) \cdot (LP_i^S(a,b))_l^B$$
$$Q_{S_i} = \max QA_{S_i}, QB_{S_i}$$

Тоді для будь-яких a, $b \neq 0$ *виконується наступна рівність:*

$$MELP^{[1..2]}(a,b) \leqslant max_i Q_{S_i}$$

Отже, максимальне значення Q_{S_i} задає верхню границю для максимального очікуваного лінійного потенціалу. Алгоритм для оцінки нижньої границі буде наведено у другому розділі.

Висновки до розділу 1

У даному розділі було введено деякі поняття лінійного криптоаналізу та обґрунтування їх важливості для оцінки стійкості до лінійного криптоаналізу. Також проведено огляд варіантів узагальнення лінійного криптоаналізу для застосування до небінарних шифрів. Введені означення будуть використовуватись у подальшому аналізі і застосуванні алгоритмів для оцінки стійкості шифрів до узагальненого лінійного криптоаналізу. Також введено поняття SP-мережі та наведено опис алгоритму шифрування Midori, для якого надалі будуть обраховані оцінки для максимального середнього лінійного потенціалу.

2 ОБЧИСЛЕННЯ ОЦІНОК ЗНАЧЕННЯ $MELP$

В даному розділі наведено детальний опис алгоритму для обчислення нижньої межі для $MELP$. Також приведено експериментальні розрахунки цієї характеристики для двох раундів шифру Midori64 з використанням цього алгоритму для варіацій небінарних лінійних перетворень та бінарного перетворення цього шифру.

2.1 Алгоритм оцінювання стійкості SP-мереж до узагальненого лінійного криптоаналізу

В цьому розділі наведемо алгоритм для обчислення нижньої границі значення MELP для 2 раундової SP-мережі та його обгрунтування.

Пропустимо лінійну трансформацію на другому раунді. Таким чином, активні S-блоки на другому раунді можуть бути визначені прямо з вихідної маски на цьому раунді, без застосування леми 1.1.

Нехай a, $b \in \{1,0\}^n$ вхідні та вихідні маски, відповідно, для першого та другого раунду. Нехай $f = wt(\gamma_a)$, $l = wt(\gamma_b)$. Пронумеруємо активні S-блоки на першому раунді як $S_1^1, ...S_f^1$ та S_1^2,...S_l^2 – на другому.

Також позначимо α_i – маски для входу на S-блок S_i^1, та β_j – для S_j^2. Характеристики з $ALH(a,b)$ матимуть форму $\langle a, y, b\rangle$. Пронумеруємо середні маски в характеристиках як $y_1, y_2, ...y_W$, $W = W_l[\gamma_a,\gamma_b]$. Значення y_w є вхідною маскою для другого раунду.

Позначимо відповідні вихідні маски нелінійного шару першого раунду як $x_1, x_2, ...x_W$ (x_w і y_w пов'язані співвідношенням з леми 1.1).

Для деякого x_w позначимо $\chi_{(w,i)}$ вихідну маску S_i^1, і для y_w – $\nu_{(w,j)}$ вхідну маску для S_j^2. Тоді з теореми 1.1 випливає рівність:

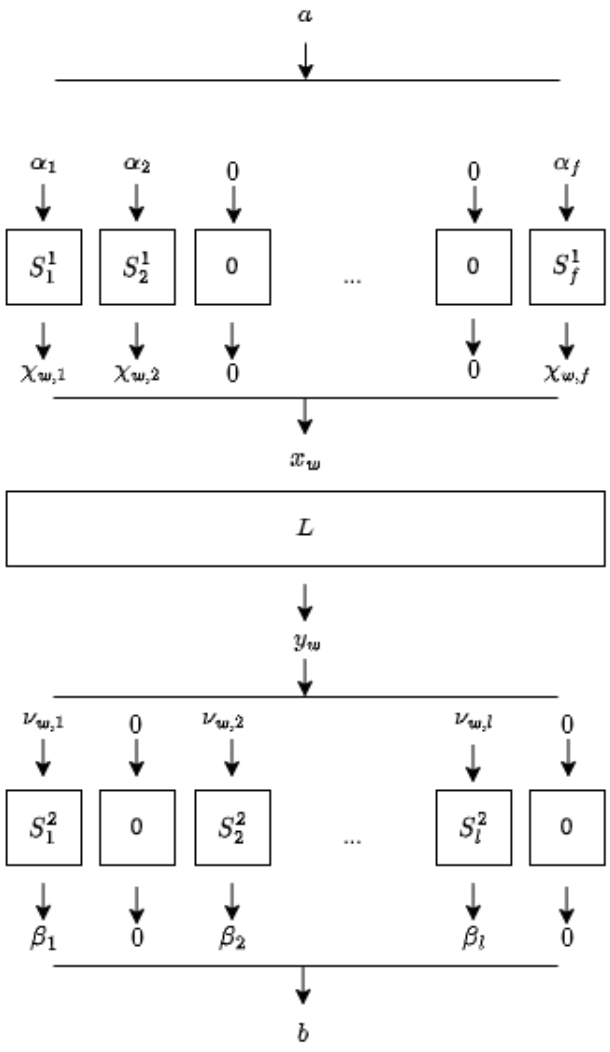

Рисунок 2.1 – Схема позначень на прикладі одного шаблону активних S-блоків

$$ELP^{[1..2]}(a,b) = \sum_{w=1}^{W}(\prod_{i=1}^{f} LP^{S_i^1}(\alpha_i, \chi_{(w,i)}) \cdot \prod_{j=1}^{l} LP^{S_j^2}(\nu_{(w,j),\beta_j})) \qquad (2.1)$$

Зручно обєднати значення масок в вектор

$$V_w = \langle \chi_{(w,1)}, \chi_{(w,2)}, ..., \chi_{(w,f)}, \nu_{(w,1)}, \nu_{(w,2)}, ..., \nu_{(w,l)} \rangle$$

Лема 2.1. *Нехай* $a, b \in \{1,0\}^n$ *такі що* $wt(\gamma_a) + wt(\gamma_b) = B_l$. *Нехай* $W = W_l[\gamma_a, \gamma_b], f = wt(\gamma_a), l = wt(\gamma_b)$. *Тоді для фіксованого* $1 \leqslant i \leqslant f$

значення $\chi_{(1,i)}, ...\chi_{(W,i)}$ *– різні і для фіксованого* $1 \leqslant j \leqslant l$ *значення* $\nu_{(1,j)}, ..., \nu_{(W,j)}$ *теж різні. Іншими словами, для векторів* $V_{w\,w=1}^{W}$ *всі значення в будь-якій позиції різні*

Надалі значення $\chi_{(w,i)}, \nu_{(w,j)}$ будуть залежати лише від значень γ_a, γ_b, а не a, b.

Таким чином, позначимо масив $B_l - list$ як масив масивів $V_{w=1}^{W}$ які сформовані вибором таких a, b що $wt(\gamma_a) + wt(\gamma_b) = B_l$, тобто, які відповідають певному шаблону активних S-блоків $s = (\gamma_a, \gamma_b)$ і задовільняють лемі 2.1

$$B_l - list = \{Z_s = \{V_w = \langle \chi_{(w,i)},...,\chi_{(w,i)},\nu_{(w,1)},...,\nu_{(w,l)} \rangle, 1 \leqslant w \leqslant W_s\}\}$$

Будемо позначати $\delta(Z_s) = W_s$ – кількість векторів в масиві. Для кожного вектору $z = \langle z_1, z_2,...z_{B_l} \in Z_s \rangle$ кожна координата є або вихідною маскою S-блоку на першому раунді, або вхідною маскою S-блоку на другому раунді.

У першому випадку, позначимо $LP^*(\alpha_i, z_i) = LP(\alpha_i, z_i)$, в другому – $LP^*(\alpha_i, z_i) = LP(z_i, \alpha_i)$

Означення 2.1. Нехай $Z \in B_l - list$. Тоді

$$\sigma(Z) = max_{\alpha_1,...\alpha_{B_l} \in \{1,0\}^n 0}(\sum_{\langle z_1,...z_{B_l} \in Z \rangle} \prod_{i=1}^{B_l} LP^*(\alpha_i, z_i))$$

Теорема 2.1. *Двораундовий MELP обмежений знизу значенням*

$$max\sigma(Z) : Z \in B_l - list$$

Доведення. Дане твердження випливає з 2.1 та визначення MELP, оскільки воно еквівалентне

$$max_{a,b \in \{1,0\}^n \ 0; wt(\gamma_a)+wt(\gamma_b)=B_l} ELP^{[1..2]}(a, b)$$

Отже, двораундовий MELP обмежений знизу максимальним значенням $\sigma(Z)$

Алгоритм 2.1. **procedure** LOWERBOUNDMELPMIDORI5(B_l, $B_l - list$, $LP(*,*)$)

lowerBound := 0
for $s = 1$ to $C_N^{B_l}$ **do**
  $Z = B_l - list[s]$
  for $1 \leqslant \alpha_1, ..., \alpha_{B_l} \leqslant 2^N$ **do**
    sum:=0
    for $w = 1$ to $\delta(Z_s)$ **do**
      prod := 1
      $V_w = Z[w] = \langle z_1,...,z_{B_l} \rangle$
      for $i = 1$ to B_l **do**
        prod = prod $\times$ $LP^*(\alpha_i, z_i)$
      end for
      sum = sum + prod
    end for
    if sum $>$ lowerBound **then**
      lowerBound = sum
    end if
  end for
end for
end procedure

2.2 Нижні границі двораундового MELP для шифру Midori

Midori64 має три варіанти лінійного перетворення, які задаються матрицями M_A, M_B, та M_C, з яких перші два мають складнішу форму та індекс розгалуження $B_l = 5$, а третя матриця, суттєво простіша, описує

двійкове перетворення із індексом розгалуження 4. Перші дві матриці можна розглядати як небінарні перетворення над кільцем лишків за модулем 16, що дозволяє перетворити шифр Midori64 у небінарний. Для таких версій Midori актуальним є саме узагальнений лінійний криптоаналіз. При цьому треба зауважити, що індекс розгалуження даних матриць над кільцем лишків також знизиться до значення $B_l = 4$. Застосувавши викладену у попередній секції методику, було отримано оцінки для нижньої границі узагальненого середнього лінійного потенціалу дворaундових небінарних версій шифру Midori64, які наведені в таблиці 2.1.

Також приведемо значення верхніх границь, отримані аналітично у таблиці 2.2.

Можна побачити, що для M_C верхня і нижня границі збігаються, а отже це і є точне значення MELP. Щодо M_A, M_B, які є небінарними, значення відрізняються на порядок, що означає що вони можуть бути значно покращені.

Для гарантованої оцінки стійкості зазвичай обирають верхню границю, оскільки вона визначає нижню межу для складності проведення атаки. Однак можна побачити, що значення границь відрізняються на порядок. Потенційно це може означати, що точне значення MELP може бути суттєво менше за верхню границю, а відповідні оцінки стійкості до узагальненого лінійного криптоаналізу – значно покращені. Також треба

Таблиця 2.1 – Нижня границя дворaундового MELP Midori64

Лінійне перетворення	Нижня границя MELP	B_l
M_A	0.02498233	4
M_B	0.02140561	4
M_C	0.015625	4

Таблиця 2.2 – Верхня границя MELP двораундового Midori64

Тип лінійного перетворення Midori64	Верхня границя MELP
небінарний	0.11392356
бінарний	0.015625

зауважити, що для бінарної версії Midori64 із перетворенням M_C, яка й була прийнята як основна версія даного шифру, наведена методика дає однакову верхню та нижню границю для MELP: 0.015625 (таким чином, це буде точне значення даної величини). Бачимо, що бінарна версія Midori64 менш уразлива до класичного лінійного криптоаналізу, аніж небінарна – до узагальненого. Це можна пояснити, в першу чергу, відмінностями у структурі класичних та узагальнених лінійних потенціалів та особливостями базової алгебраїчної структури: при наявності елементів, порядки яких більше 2 (що неможливо у двійкових шифрах) аналітик може відстежувати більше проміжних даних. Також зауважимо, що якби небінарні варіанти матриць M_A, M_B мали оригінальний індекс розгалуження рівний пяти, то оцінки MELP лежали б в діапазоні 0.008-0.03, тобто були б на порядок меншими за наведені у таблиці 2.1 та щонайменше вдвічі менші за MELP для двійкової версії шифру. Це зайвий раз підкреслює важливість індексу розгалуження для гарантування стійкості до усіх видів лінійних атак.

Висновки до розділу 2

Отже, в цьому розділі описано алгоритм для оцінки знизу значень $MELP$, а також деталі його реалізації та результати його застосування. Отримано нижні межі для значення $MELP$ для небінарних варіацій шифру Midori.

ВИСНОВКИ

Було проведено огляд опублікованих джерел за тематикою оцінок стійкості лінійного криптоаналізу, а також методів узагальнення лінійного криптоаналізу для застосування до небінарних SP-мереж.

Описано алгоритм обчислення нижньої межі максимального середнього лінійного потенціалу для дворaундової SP-мережі. Зроблено узагальнення для застосування до небінарних SP-мереж. Також його було реалізовано і застосовано для отримання оцінок стійкості шифру Midori, в результаті чого отримано нижні границі значень MELP, які рівні 0.02498233 та 0.02140561 для першої і другої варіації лінійного перетворення шифру. Враховуючи, що обчислена верхня межа MELP для двох варіацій дорівнює 0.11392356, тобто на порядок більша за отримані нижні, це свідчить про те що точне значення MELP може бути суттєво менше за верхню границю, а відповідні оцінки стійкості до узагальненого лінійного криптоаналізу потенційно значно покращені.

ПЕРЕЛІК ПОСИЛАНЬ

1. Lars R. Knudsen, Matthew J. B. Robshaw. The Block Cipher Companion. Springer, 2011.

2. Subhadeep Banik, Andrey Bogdanov, Takanori Isobe, Kyoji Shibutani, Harunaga Hiwatari, Toru Akishita, and Francesco Regazzoni Midori: A Block Cipher for Low Energy (Extended Version) 2015 — Режим доступу: https://eprint.iacr.org/2015/1142.pdf

3. Matsui M. A New Method for Known Plaintext Attack of FEAL Cipher / M. Matsui, A. Yamagishi // Advances in Cryptology — EUROCRYPT'92. – Lecture Notes in Computer Science. – vol. 658. – Springer-Verlag, 1993. – pp. 81–91.

4. Matsui M. Linear cryptanalysis methods for DES cipher / M. Matsui // Advances in Cryptology – EUROCRYPT'93, Proceedings. – Springer Verlag, 1994. – pp. 386–397.

5. Harpes C. A Generalization of Linear Cryptanalysis and the Applicability of Matsui's Piling-up Lemma / C. Harpes, G.G. Kramer, J.L. Massey // Advances in Cryptology – EUROCRYPT'95, Proceedings. – Springer Verlag, 1995. – pp. 24–38.

6. Courtois N.T. Feistel schemes and bi-linear cryptanalysis / N.T. Courtois // Advances in Cryptology – CRYPTO'04, Proceedings. – Springer Verlag, 2004. – pp. 23–40.

7. Алексейчук А. Верхние границы максимальных значений средних вероятностей дифференциальных и линейных характеристик шифра Фейстеля, содержащего сумматор по модулю 2n. / А. Алексейчук, Л. Ковальчук // Прикладная радиоэлектроника. – 2005. – т.5, No1. – С. 74-82.

8. Baignères T., Stern J., Vaudenay S. Linear Cryptanalysis of Non Binary Ciphers Режим доступу: https://www.di.ens.fr/~stern/data/St122.pdf

9. Howard M. Heys A Tutorial on Linear and Differential Cryptanalysis — Режим доступу:https://ioactive.com/wp-content/uploads/2015/07/ldc_

tutorial.pdf

10. Hong S. Provable security against differential and linear cryptanalysis for the SPN structure / S. Hong, S. Lee, J. Lim, J. Sung, D. Cheon // Fast Software Encryption. – FSE'00, Proceedings. – Springer Verlag, 2001. – P. 273 – 283.

11. Kang J.-S. Practical and provable security against differential and linear cryptanalysis for substitution-permutation networks / J.-S. Kang, S. Hong, S. Lee, O. Yi, C. Park, J. Lim // ETRI Journal. – 23. – 2001. – pp. 158-167.

12. Keliher L. New method for upper bounding the maximum average linear hull probability for SPNs / L. Keliher, H. Meijer, and S. Tavares // Lecture Notes in Computer Science. – vol. 2045. – Berlin: Springer, 2001. – pp. 420-436.

13. Keliher L. Improving the upper bound on the maximum average linear hull probability for Rijndael / L. Keliher, H. Meijer, and S. Tavares // Eighth Annual International Workshop on Selected Areas in Cryptography (SAC 2001). – LNCS, vol. 2259. – Springer-Verlag, 2001. – pp. 112–128.

14. Nyberg K. Provable Security Against a Differential Attack / K. Nyberg, L.R. Knudsen // Journal of Cryptology. – Vol.8. – No.1. – 1995.

ДОДАТОК А ТЕКСТИ ПРОГРАМ

```
ppackage ua.kpi;

import org.apache.commons.lang3.StringUtils;

import java.util.*;
import java.util.stream.Stream;

public class LowerMEDPBound {

    public static void performMidori4(int n, int[] sbox, int[]
        int branchNumber = 4;
        var Blists = BlistsMidori(matrix, branchNumber, binary
                .entrySet().stream().toList();
        var lowerBound = 0d;
        for (int s = 0; s < Blists.size(); s++) {
            var blist = Blists.get(s).getValue();
            var pattern = Blists.get(s).getKey();
            var patternStr = StringUtils.leftPad(Integer.toBin
            var patternsBitsNumbers = new ArrayList<Integer>()
            for(int i=0;i<8;i++){
                if(patternStr.charAt(i) == '1'){
                    patternsBitsNumbers.add(i);
                }
            }
            for (int a1 = 1; a1 < 16; a1++) {
                for (int a2 = 1; a2 < 16; a2++) {
                    for (int a3 = 1; a3 < 16; a3++) {
                        for (int a4 = 1; a4 < 16; a4++) {
                            var sum = 0d;
```

```
                        for (List<Integer> blistW : blist)
                            sum = sum + lpTable[patternsBit
                                    [patternsBitsNumbers.g
                                    * lpTable[patternsBitsN
                                    [patternsBitsNumbers.g
                                    * lpTable[patternsBitsN
                                    [patternsBitsNumbers.g
                                    * lpTable[patternsBitsN
                                    [patternsBitsNumbers.g
                        }
                        if (sum > lowerBound) {
                            lowerBound = sum;
                        }
                    }
                }
            }
        }
        System.out.println("LOWER_BOUND_" + lowerBound);
    }

    public static double[][] LPTableNormNonBinary(int n, int[]
        var nSize = 1 << n;
        var lpTable = new double[nSize][nSize];
        for (int a = 0; a < nSize; a++) {
            for (int b = 0; b < nSize; b++) {
                var sinsum = 0.;
                var cossum = 0.;
                for (int x = 0; x < nSize; x++) {
                    var arg = 2 * Math.PI * (((a * x) % nSize)
                            ((b * sbox[x]) % nSize)) / nSize;
```

```
                sinsum = sinsum + Math.sin(arg);
                cossum = cossum + Math.cos(arg);
            }
            lpTable[a][b] = (Math.pow(sinsum, 2) + Math.pow
        }
    }
    return lpTable;
}


private static int[] mixColumnsMidori(int[] state, int[][]
    int temp0, temp1, temp2, temp3;

    temp0 = mult(matrix[0][0], state[0]) ^ mult(matrix[0][1
            mult(matrix[0][2], state[2]) ^ mult(matrix[0][3
    temp1 = mult(matrix[1][0], state[0]) ^ mult(matrix[1][1
            mult(matrix[1][2], state[2]) ^ mult(matrix[1][3
    temp2 = mult(matrix[2][0], state[0]) ^ mult(matrix[2][1
            mult(matrix[2][2], state[2]) ^ mult(matrix[2][3
    temp3 = mult(matrix[3][0], state[0]) ^ mult(matrix[3][1
            mult(matrix[3][2], state[2]) ^ mult(matrix[3][3

    state[0] = temp0;
    state[1] = temp1;
    state[2] = temp2;
    state[3] = temp3;

    return state;
}

private static int[] mixColumnsMidoriNonBinary(int[] state
```

```
        int temp0, temp1, temp2, temp3;

        temp0 = (matrix[0][0] * state[0] + matrix[0][1] * state
                 matrix[0][2] * state[2] + matrix[0][3] * state
                 % state.length;
        temp1 = (matrix[1][0] * state[0] + matrix[1][1] * state
                 matrix[1][2] * state[2] + matrix[1][3] * state
                 % state.length;
        temp2 = (matrix[2][0] * state[0] + matrix[2][1] * state
                 matrix[2][2] * state[2] + matrix[2][3] * state
                 % state.length;
        temp3 = (matrix[3][0] * state[0] + matrix[3][1] * state
                 matrix[3][2] * state[2] + matrix[3][3] * state
                 % state.length;

        state[0] = temp0;
        state[1] = temp1;
        state[2] = temp2;
        state[3] = temp3;

        return state;
    }

    private static int mult(int a, int b) {
        int sum = 0;
        while (a != 0) {
            if ((a & 1) != 0) {
                sum = sum ^ b;
            }
            b = times(b);
            a = a >>> 1;
```

```
        }
        return sum;

    }

    private static int times(int b) {
        if ((b & 0x80) == 0) {
            return b << 1;
        }
        return (b << 1) ^ 0x11b;
    }


    public static List<Integer> subset5from8() {
        var res = new ArrayList<Integer>();
        for (int i = 0; i < 6; i++) {
            for (int j = i + 1; j < 7; j++) {
                for (int m = j + 1; m < 8; m++) {
                    res.add(0b11111111 - (1 << i) - (1 << j) -
                }
            }
        }
        return res;
    }

    public static List<Integer> subset4from8() {
        var res = new ArrayList<Integer>();
        for (int i = 0; i < 5; i++) {
            for (int j = i + 1; j < 6; j++) {
                for (int m = j + 1; m < 7; m++) {
                    for (int n = m + 1; n < 8; n++) {
```

```
                        res.add(0b11111111 - (1 << i) - (1 << j
                                - (1 << m) - (1 << n));
                    }
                }
            }
        }
        return res;
    }

    public static Map<Integer, List<List<Integer>>> BlistsMido
            int[][] matrix, int branchNumber, boolean binary) {

        Map<Integer, List<List<Integer>>> Blists = new HashMap<

        int block1, block2, block3, block4, block5, block6, blo
                firstRoundActiveSboxes, secondRoundActiveSboxes
        var patterns = branchNumber == 5 ? subset5from8() : sub
        patterns.forEach(pattern -> Blists.put(pattern, new Ar
        for (int i = 0; i < 16; i++) {
            System.out.println("i " + i);
            for (int j = 0; j < 16; j++) {
                for (int k = 0; k < 16; k++) {
                    for (int m = 0; m < 16; m++) {
                        var x = new int[]{i, j, k, m};

                        block1 = isActive(i);
                        block2 = isActive(j);
                        block3 = isActive(k);
                        block4 = isActive(m);

                        firstRoundActiveSboxes = block1 + block
```

```
                    if(binary){
                        mixColumnsMidori(x, matrix);
                    } else {
                        mixColumnsMidoriNonBinary(x, matrix
                    }

                    block5 = isActive(x[0]);
                    block6 = isActive(x[1]);
                    block7 = isActive(x[2]);
                    block8 = isActive(x[3]);

                    secondRoundActiveSboxes = block5 + bloc

                    if (firstRoundActiveSboxes + secondRou
                        int pattern = (block1 << 7) + (bloc
                                + (block3 << 5) + (block4 <
                                + (block5 << 3) + (block6 <
                                + (block7 << 1) + block8;
                        var value = Stream.of(i, j, k, m, x
                                .filter(v -> v != 0).toLis
                        Blists.get(pattern).add(value);
                    }

                }
            }
        }
        return Blists;
    }
```

```
    public static double[][] LPTableNorm(int n, int[] sbox) {
        var nSize = 1 << n;
        var lpTable = new double[nSize][nSize];

        for (int a = 0; a < nSize; a++) {
            for (int b = 0; b < nSize; b++) {
                for (int x = 0; x < nSize; x++) {
                    lpTable[a][b] = lpTable[a][b] +
                            ((scalarMul(a, x) ^ scalarMul(b, sb
                }
                lpTable[a][b] = Math.pow(lpTable[a][b] / nSize
            }
        }
        return lpTable;
    }

    public static int scalarMul(int a, int b) {
        return Integer.bitCount(a & b) & 1;
    }

    private static int isActive(long value) {
        return (value == 0 ? 0 : 1);
    }

}
```