КАЗАНСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ

Ш.Т. ИШМУХАМЕТОВ, Б.Г. МУБАРАКОВ

ЭЛЕМЕНТЫ ЛИНЕЙНОГО КРИПТОАНАЛИЗА

Учебно-методическое пособие

КАЗАНЬ
2025

УДК 004.056
ББК 16.8
И97

Рекомендовано к изданию
Учебно-методической комиссией ИВМиИТ
(протокол № 1 от 19 сентября 2025 года)

Рецензенты:
кандидат физико-математических наук, доцент **Байрашева В.Р.**
доктор физико-математических наук, профессор **Аблаев Ф.М.**

Ишмухаметов Ш.Т.
И97 Элементы линейного криптоанализа: учебно-методическое пособие / Ш.Т. Ишмухаметов, Б.Г. Мубараков. – Казань: Казан. ун-т, 2025. – 34 с.

Данное учебно-методическое пособие содержит три лабораторные работы для углубленного изучения метода линейного криптоанализа блочных шифров и хеш-функций. Практические задания сопровождается теоретическим материалам по построению блочных шифров на основе схемы Фейстеля.

В ходе выполнения работ студенты получают знания по разработке блочных шифров и хеш-функций, устойчивых к атакам с использованием линейного криптоанализа.

Предназначено для студентов старших курсов и магистрантов, специализирующихся по направлению «Информационная безопасность».

УДК 004.056
ББК 16.8

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Существует два вида симметричных шифров, используемых в криптографии – это потоковые и блочные шифры. Первые шифруют поток информации посимвольно, складывая побайтно или побитово информационный поток с ключевой последовательностью. Блочные шифры разбивают текст на блоки определенной длины (обычно, 64, 128 и более бит) и используют схему Фейстеля или подобные для перестановок и смешивания ключей внутри каждого блока в ходе много раундовых процедур.

В свою очередь, для потоковых шифров ключевая последовательность строится либо, как одноразовый блокнот, когда при каждом шифровании заново используется новый ключ (например, построенный на фрагменте какого-то литературного произведения), либо используется генератор псевдослучайной последовательности, когда ключ небольшой длины может генерировать псевдослучайную последовательность большого периода. Ключевая последовательность, получаемая с помощью таких генераторов, должна удовлетворять ряду требований, которые позволяют успешно противостоять различным видам криптографических атак. Требования к псевдослучайным последовательностям изложены в тестах американского Института стандартов NIST [1].

Блочные шифры используют, в основном, схему Фейстеля, разработанную еще в 70-х годах XX столетия. В 1971 году Хорст Фейстель запатентовал два устройства, реализующих различные алгоритмы шифрования, позже получившие название «Люцифер». Тогда Фейстель работал над созданием новых криптосистем в компании IBM вместе с Доном Копперсмитом. Проект «Люцифер» был экспериментальным, но стал основой для алгоритма DES (англ. Data Encryption Standard).

Любой блочный шифр разбивает шифруемый файл на блоки фиксированной длины (например, для DES он был равен 64 битам) и выполняет многораундовые преобразования, состоящие из однотип-

ных преобразований, включающих подстановки, перестановки и гаммирование. Гаммирование – это сложение блока данных с ключевой последовательностью, такое же, как в потоковых шифрах.

Преобразование называется линейным, если оно может быть выражено с помощью набора линейных функций $f_j(x_1, x_2, \dots, x_n), j = 1,2, \dots k$. Линейными называются функции вида $f(x_1, x_2, \dots, x_n) = a_1x_1 + a_2x_2 + \cdots + a_nx_n + b$.

Перестановки и гаммирование – это линейные преобразования, а подстановки – нелинейные. Линейный криптоанализ – это вид атаки, позволяющий находить ключ шифрования путем анализа входных и выходных блоков данных.

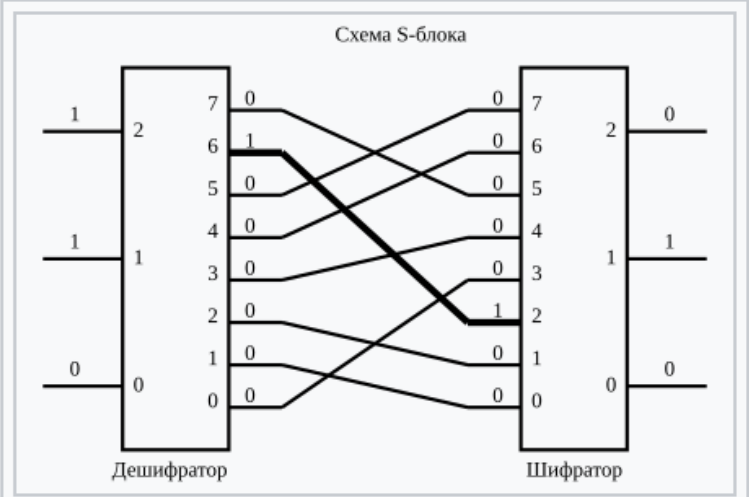
Он работает тем эффективнее, чем ближе к линейному является преобразование. Поэтому подстановки или s-боксы в блочных шифрах являются необходимым элементом. Задачей первой лабораторной работы является построение нелинейного преобразования с высокой степенью нелинейности.

ЛАБОРАТОРНАЯ РАБОТА 1. ИССЛЕДОВАНИЕ СВОЙСТВ S-БОКСОВ И ПОСТРОЕНИЕ S-БОКСА НАИБОЛЬШЕЙ СТЕПЕНИ НЕЛИНЕЙНОСТИ

1.1. Теоретический материал

Обозначим через F_2 множество $\{0,1\}$, а через F_2^n множество векторов длины n с элементами из F_2.

Произвольный s-бокс длины (n, m) представляет собой отображение из F_2^n в F_2^m. Образом отображения служит все множество F_2^m, то есть s-box является отображением *на*. При $n = m$ также требуют отсутствия неподвижных точек, то есть для всех аргументов x, $s(x) \neq x$ (рис. 1):



Принципиальная схема трёхразрядного ($n = 3$) s-блока

Рис.1. Пример s-бокса при $(m, n) = (3,3)$

Вход $x = 5 = 101_2$ он преобразует в $y = 7 = 111_2$. Также s-бокс можно задать с помощью таблицы. Пусть, например, $n = 3, m = 2$ (рис. 2).

Самый левый бит входа ↓	00	01	10	11	← Самые правые биты входа
0	00	10	01	11	
1	10	00	11	01	

Рис. 2. S-бокс (3, 2)

Тогда, $S(100) = 11$ (первый бит аргумента 100 определяет строку, а последние два бита – столбец таблицы).

Произвольный (n, m) s-бокс представляет собой набор из m булевых функций размерности n. Напомним основные определения и обозначения из теории булевых функций. $F_2 = \{0,1\}$, F_2^n – множество n-мерных векторов с координатами из F_2. Каждую булеву функцию можно задать в виде полинома Жегалкина:

$$f = a_0 + \sum a_k x_1^{k_1} x_2^{k_2} \dots x_n^{k_n},$$

где $k_i \in F_2 = \{0,1\}$, и $x_i^0 = 1,\ x_i^1 = x_i$. Значит, булеву функцию в виде полинома Жегалкина можно задать как набор векторов $w_i = (k_1, k_2, \dots k_n)$ и свободного члена $a_0 \in F_2$.

Пример. Пусть $f = 1 + x_1x_3 + x_2x_4$. Эту функцию можно задать как список:

$$F = [1, (1010), (0101)].$$

Функция f называется ***линейной*** (или аффинной), если все слагаемые в полиноме Жегалкина имеют первую степень, т.е. $f = a_0 + \sum a_i x_i = a_0 + AX$, где $A = (a_1, a_2, a_3),\ X = (x_1, x_2, x_3)$.

Пример: $f(x_1, x_2, x_3) = 1 + x_1 + x_3$, $a_0 = 1,\ A = (1,0,1)$.

Булеву функцию можно также задать как вектор ее значений. Пусть, например, $f(x_1, x_2, x_3) = x_1x_2 + x_3$. Построим таблицу истинности (таблица 1) для функции f:

Множество всех линейных функций размерности n обозначается через LF_n.

Таблица 1

Таблица истинности

(x_1, x_2, x_3)	(0,0,0)	(0,0,1)	(0,1,0)	(0,1,1)	(1,0,0)	(1,0,1)	(1,1,0)	(1,1,1)
$f(x_1, x_2, x_3)$	0	1	0	1	0	1	1	0

$F = (0,1,0,1,0,1,1,0)$.

Нормой Хэмминга вектора $a \in F_2^n$ называется число его ненулевых координат. Например, $H(101) = 2$.

Расстоянием Хэмминга между двумя векторами называется норма их суммы (при сложении векторы складываются покоординатно). По-другому, расстояние Хэмминга равно числу координат, в которых эти вектора отличаются. Например, $H(1010, 1100) = H(1010 \oplus 1100) = H(0110) = 2$.

Расстояние между двумя n-мерными функциями f и g вычисляется по формуле:

$$\rho(f, g) = |x: x \in F_2^n, f(x) \neq g(x)|$$

Пример. Найдем расстояние между функциями двух переменных – логическими «и» и «или». Зададим их таблицы истинности (таблица 2).

Таблица 2

Таблица истинности для «и» и «или»

(x_1, x_2)	«и»	«или»	«и» + «или»
(0, 0)	0	0	0
(0, 1)	0	1	1
(1, 0)	0	1	1
(1, 1)	1	1	0

Расстояние между «и» и «или» равно числу ненулевых аргументов в их сумме, т.е. равно 2: $\rho(\&, \vee) = 2$.

Мерой линейности булевой функции f называется наименьшее возможное расстояние от f до линейной функции:

$$NL(f) = \min_g\{\rho(f, g), g \in LF_n\}.$$

Пример. $n = 2$. Выпишем всевозможные линейные функции размерности 2:

$$f_1(x_1, x_2) = 0, \qquad f_2(x_1, x_2) = 1,$$
$$f_3(x_1, x_2) = x_1, \qquad f_4(x_1, x_2) = x_2,$$
$$f_5(x_1, x_2) = x_1 + 1, \qquad f_6(x_1, x_2) = x_2 + 1,$$
$$f_7(x_1, x_2) = x_1 + x_2, \qquad f_8(x_1, x_2) = x_1 + x_2 + 1.$$

Составим для них таблицы истинности (таблица 3):

Таблица 3

Таблица истинности для функций двух переменных

x_1	x_2	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8
0	0	0	1	0	0	1	1	0	1
0	1	0	1	0	1	1	0	1	0
1	0	0	1	1	0	0	1	1	0
1	1	0	1	1	1	0	0	0	1

Всего существует 2^{2^n} функций от n переменных. Для $n = 2$ это число равно 16, значит, существует 8 нелинейных функций. Каждая из них является суммой логического «и» $f = x_1 x_2$ и какой-то линейной функции.

Функция $f = x_1 x_2$ задается вектором:

$$f = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}.$$

Очевидно, наименьшее расстояния между f и линейными функциями равно 1 (это расстояние до нулевой функции, столбец значений которой состоит из 0).

Если n – четно, то максимальное значение $NL(f)$ равно $2^{n-1} - 2^{\frac{n}{2}-1}$. Функции с нелинейностью $2^{n-1} - 2^{\frac{n}{2}-1}$ называются bent-функциями. Для нечетных n bent-функций не существует. Но можно найти функцию с максимальной нелинейностью.

Примером bent-функции при n = 6 является $f = x_1x_3 + x_2x_4$. Ее степень нелинейности равна 4. Значения максимальной нелинейности четных n приведено в таблице 4:

Таблица 4

Значения максимальной нелинейности четных n

n	2	4	6	8	10
maxNL	1	6	28	120	496

Таким образом, при увеличении n максимальная возможная степень быстро растет. Построение bent-функций для больших n является очень трудной задачей.

1.2. Степень нелинейности s-бокса

Произвольный s-box представляет собой набор из базовых функций $f_1, f_2, \dots f_m$. Степень нелинейности s-бокса определяется как наименьшее расстояние между множеством всех линейных комбинаций этих базовых функций и множеством всех линейных функции данной размерности.

Значит, для определения степень нелинейности s-бокса надо строить всевозможные линейные комбинации функций $f_1, f_2, \dots f_m$ и находить расстояние от этих линейных комбинаций до линейных функций.

Пример. Пусть n = 3, и s-бокс задан циклической перестановкой сдвига вправо на 1 позицию:

$$(x_1, x_2, x_3) \to (x_2, x_3, x_1)$$

Зададим этот s-бокс с помощью таблиц истинности (таблица 5).

Такой s-бокс имеет нулевую степень нелинейности, поскольку все его базовые функции просто проекции:

$$f_1(x_1, x_2, x_3) = x_2,\ \ f_2(x_1, x_2, x_3) = x_3, f_3(x_1, x_2, x_3) = x_1.$$

Поэтому для построения нелинейного s-бокса нельзя использовать перестановки типа сдвига, а надо определять s-бокс либо через базовые нелинейные функции, либо перестановки, не являющиеся сдвигом. Иначе говоря, надо задать n нелинейных базовых функций и

проверить, что их линейные комбинации $\sum a_i f_i$ для произвольных векторов $A = (a_1, a_2, \dots a_n)$ также являются нелинейными.

Таблица 5

S-бокс размерности 3

(x_1, x_2, x_3)	$S(x_1, x_2, x_3)$	f_1	f_2	f_3
(0,0,0)	(0,0,0)	0	0	0
(0,0,1)	(0,1,0)	0	1	0
(0,1,0)	(1,0,0)	1	0	0
(0,1,1)	(1,1,0)	1	1	0
(1,0,0)	(0,0,1)	0	0	1
(1,0,1)	(0,1,1)	0	1	1
(1,1,0)	(1,0,1)	1	0	1
(1,1,1)	(1,1,1)	1	1	1

Для $n = 3$ существует 7 нетривиальных линейные комбинации этих функций:

$$f_1, f_2, f_3, f_1+f_2, f_1+f_3, f_1+f_3, f_1+f_2+f_3$$

и еще один такой же набор с добавлением 1 к каждой из этих функций, всего 14. Однако поскольку степень нелинейности функций f и $f + 1$ одинаковая (докажите это!), то достаточно рассматривать линейные комбинация со свободным членом 0.

Пример. Попробуйте построить функцию от трех переменных, имеющую степень нелинейности больше или равную 2.

1.3. Задание на лабораторную работу 1

1) Разработать программу, которая находит расстояние Хэмминга между булевыми функциями и определяет степень нелинейности заданной булевой функции размерности n как минимальное расстояние до произвольной линейной (аффинной) функции.

2) Разработать программу, строящую bent-функции размерностей $n = 4$ и $n = 6$.

3) Разработать программу, генерирующую нелинейные s-боксы и с помощью нее построить s-бокс размерности $n = 5$ с наибольшей степенью нелинейности.

ЛАБОРАТОРНАЯ РАБОТА 2. ПОСТРОЕНИЕ ТАБЛИЦЫ ЛИНЕЙНОЙ АППРОКСИМАЦИИ S-БОКСА

2.1. Теоретический материал

Как было сказано ранее, произвольный s-бокс представляет собой отображение из пространства векторов размерности n с элементами из $Z_2 = \{0,1\}$ в пространство векторов размерности m с такими же элементами с определенными свойствами. S-бокс может быть задан либо как перестановка векторов, либо как набор m булевых функций от n-переменных. Каждый s-бокс характеризуется своей ***степенью нелинейности***, которая определяется как наименьшее расстояние по Хэммингу между линейными комбинациями булевых функций, образующих s-бокс, и классом линейных булевых функций. Чем выше степень нелинейности s-бокса, тем успешнее s-бокс противостоит крипто атакам с использованием нелинейного криптоанализа.

В этой лабораторной работе мы изучим процедуру построения таблицы линейно аппроксимации (linear approximation table LAT) для заданного s-бокса, которая используется для выполнения атак нелинейного криптоанализа. Будем изучать необходимые понятия на примере s-бокса размерности 3x3. Предположим, что нам задан следующий s-бокс (таблица 6):

Таблица 6

S-бокс

X	(0,0,0)	(0,0,1)	(0,1,0)	(0,1,1)	(1,0,0)	(1,0,1)	(1,1,0)	(1,1,1)
S(X)	(1,0,0)	(1,0,1)	(1,1,0)	(0,0,1)	(1,1,1)	(1,0,0)	(1,0,0)	(0,0,1)

Данный s-бокс можно также представить как набор из 3-х булевых функций, заданных таблицей истинности. Они называются базисными функциями s-бокса (таблица 7).

Кратко запишем базисные функции в виде строк их значений:

$Y_1 = [1,1,1,0,1,0,1,0]$, $Y_2 = [0,0,1,0,1,1,0,0]$, $Y_3 = [0,1,0,1,1,1,0,1]$.

Таблица 7

Задание с помощью базисных функций

X_1	X_2	X_3	$f_1(X)$	$f_2(X)$	$f_3(X)$
0	0	0	1	0	0
0	0	1	1	0	1
0	1	0	1	1	0
0	1	1	0	0	1
1	0	0	1	1	1
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	0	0	1

Линейной аппроксимацией s-бокса называется произвольное соотношение типа:

$$\sum a_i X_i = \sum b_1 Y_i + c\,, \qquad (1)$$

где X_i, Y_j – вектора размерности n, а $+$ это операция XOR (сложение по модулю 2). Подставляя в (1) всевозможные комбинации аргументов X_i мы определим, как часто выполняется заданная линейная аппроксимация. Построим, например, таблицу значений для соотношения $X_2 = Y_1 + Y_3$ (таблица 8).

Таблица 8

Таблица значений для соотношения $X_2 = Y_1 + Y_3$

X_1	X_2	X_3	Y_1	Y_2	Y_3	$Y_1 + Y_3$	X_2
0	0	0	1	0	0	0	0
0	0	1	1	0	1	1	0
0	1	0	1	1	0	1	1
0	1	1	0	0	1	1	1
1	0	0	1	1	1	0	0
1	0	1	0	1	1	0	0
1	1	0	1	1	0	1	1
1	1	1	1	0	1	1	1

Для этого будем в цикле перебирать всевозможные тройки значений вектора (X_1, X_2, X_3), вычислять для них значения вектора ба-

зисных функций (Y_1, Y_2, Y_3) и проверять соотношение $X_2 = Y_1 + Y_3$. Сравнивая два последних столбца, мы видим, что соотношение $X_2 = Y_1 + Y_3$ выполняется в 6 случаях из 8, что выше среднего значения $\mathrm{c_{ср}} = 4$. Обозначим это значение через c, $0 \le c \le 8$. Оно связано с расстоянием Хэмминга h между двумя векторами соотношением $c + h = 2^n = 8$, поскольку h характеризует число компонент, где вектора различаются.

Мы можем также поискать и другие подходящие аппроксимации. Для этого составим таблицу линейной аппроксимации. Запишем произвольное линейное соотношение в векторной форме:

$$AX = BY,$$

где $X = (X_1, X_2, X_3)$, $Y = (Y_1, Y_2, Y_3)$, $A = (A_1, A_2, A_3)$, $B = (B_1, B_2, B_3)$.

Таблица линейной аппроксимации представляет собой таблицу размерности $2^n \times 2^n$, где заголовками столбцов служат комбинации векторов $A = (A_1, A_2, A_3)$, а заголовками строк – комбинации векторов $B = (B_1, B_2, B_3)$. На пересечении помещается число аргументов $X = (X_1, X_2, X_3)$, на которых выполняется соотношение $AX = BY$ минус $\mathrm{c_{ср}}$. Параметр $\mathrm{c_{ср}} = 4 = 2^n/2$ обозначает среднее значение c. Чем больше отличается значение c от среднего (в большую или меньшую стороны), тем лучше данная линейная комбинация характеризует s-бокс. Если же $c - \mathrm{c_{ср}} = \pm 4$, то данный s-бокс имеет наименьшую степень нелинейности 0, то есть является полностью линейным. Поэтому с точки зрения криптографии, идеальным является s-бокс, у которого все значения в таблице аппроксимаций близки к 0.

2.2. Построение таблицы линейной аппроксимации

1) Разместим всевозможные комбинации коэффициентов левой части линейной комбинации по горизонтали, а правой части – по вертикали.

2) На пересечении столбца (a_1, a_2, a_3) и строки (b_1, b_2, b_3) поместим значение числа совпадений соотношения $a_1X_1 + a_2X_2 + a_3X_3 = b_1Y_1 + b_2Y_2 + b_3Y_3$ минус 4. Например, на пересечении столб-

ца $(0,1,0)$ и строки $(1,0,1)$ поместим значение 6–4=2, где 6 – значение, вычисленное в предыдущей таблице.

3) Размер полученной таблицы равен $2^n x 2^m$, где n, m – размеры входа и выхода s-бокса. В нашем примере получится таблица 8x8.

Вручную заполнять такую таблицу слишком долго, поэтому реализуем эту процедуру на компьютере.

1) Зададим три базисные функции s-бокса столбцом векторов, как сделано выше в таблице истинности.

2) В двойном цикле перебираем векторы входа (a_1, a_2, a_3) и векторы выхода (b_1, b_2, b_3), каждый от (0,0,0) до (1,1,1).

3) Для каждой комбинации значений (a_1, a_2, a_3) и (b_1, b_2, b_3) открываем новый цикл по всевозможным значениям векторов X, X_2, X_3 и проверяем соотношение $a_1X_1 + a_2X_2 + a_3X_3 = b_1Y_1 + b_2Y_2 + b_3Y_3$. Не забудем, что суммирование выполняется по модулю 2.

4) Вычисляем число аргументов, на которых это соотношение выполняется. От полученного значения отнимаем 4 и помещаем в соответствующую клетку таблицы. Получится следующая таблица (таблица 9):

Таблица 9

Таблица линейной аппроксимации s-бокса

	(0,0,0)	(0,0,1)	(0,1,0)	(0,1,1)	(1,0,0)	(1,0,1)	(1,1,0)	(1,1,1)
(0,0,0)	4	0	0	0	0	0	0	0
(0,0,1)	-1	**3**	-1	-1	1	1	1	1
(0,1,0)	0	-2	0	2	2	0	2	0
(0,1,1)	-1	1	**3**	1	-1	1	-1	1
(1,0,0)	-2	-2	0	0	0	0	-2	2
(1,0,1)	-1	-1	1	1	-1	1	0	**-3**
(1,1,0)	0	2	-2	0	0	-2	-2	0
(1,1,1)	1	-1	-1	**-3**	-1	1	1	-1

Мы видим, что наш s-бокс, хотя и не является линейным, имеем степень нелинейности 1, так в таблице есть несколько значений -3 и 3, близкие к линейным ± 4 (значения 1-й строки не учитываются). Возьмем, например, значение – 3, расположенное в 3 столбце и 4

строке (выделено жирным шрифтом). Оно соответствует линейной комбинации $Y_2 + Y_3 = X_2$. Полученная аппроксимация имеет число совпадений $c = 3 + c_{\text{ср}} = 7$ со значениями s-бокса и не выполняется только на одном аргументе. Выпишем из таблицы еще два соотношения (строка 2, столбец 2) и (строка 6, столбец 8). Последнее значение содержит отрицательное значение, поэтому в уравнении добавили 1 в правой части:

$$\begin{cases} Y_2 + Y_3 = X_2 \\ Y_3 = X_3 \\ Y_1 + Y_3 = X_1 + X_2 + X_3 + 1, \end{cases}$$

или

$$\begin{cases} Y_2 + Y_3 + X_2 = 0 \\ Y_3 + X_3 = 0 \\ Y_1 + Y_3 + X_1 + X_2 + X_3 + 1 = 0. \end{cases} \quad (*)$$

Вычислим значения полученной аппроксимации на множестве всех аргументов и сравним с соответствующими значениями базисных функций s-бокса (таблица 10).

Таблица 10

Значения линейных комбинаций базисных функций

X_1	X_2	X_3	Y_1	Y_2	Y_3	$Y_2 + Y_3 + X_2$	$Y_3 + X_3$	$Y_1 + Y_3 + X_1 + X_2 + X_3 + 1$
0	0	0	1	0	0	0	0	0
0	0	1	1	0	1	**1**	0	0
0	1	0	1	1	0	0	0	**1**
0	1	1	0	0	1	0	0	0
1	0	0	1	1	1	0	**1**	0
1	0	1	0	1	1	0	0	0
1	1	0	1	1	0	0	0	0
1	1	1	1	0	1	0	0	0

Мы видим, что все три линейные комбинации только на одном из аргументов дают отличное от нуля значение, то есть исходная система базисных функций $Y_1 = [1,1,1,0,1,0,1,0]$, $Y_2 = [0,0,1,0,1,1,0,0]$, $Y_3 = [0,1,0,1,1,1,0,1]$ имеет степень нелинейности 1, то s-бокс почти линеен.

2.3. Задание на лабораторную работу 2

Построить таблицу линейных аппроксимаций согласно номеру своего варианта. Сделать отчет в формате Word, в котором привести таблицу линейных аппроксимаций, код процедуры построения таблицы и систему линейных уравнений, аппроксимирующих s-бокс.

Вариант 1

F1=(0,1,0,1,0,1,0,1,1,0,0,0,1,0,1,0)
F2=(0,1,1,1,1,0,1,0,0,1,0,1,1,0,1,0)
F3=(1,1,0,0,0,0,1,1,0,0,1,0,1,1,0,0)
F4=(0,1,0,1,1,0,1,0,1,0,1,0,0,1,0,0)

Вариант 2

F1=(0,1,0,1,1,0,1,0,1,0,1,0,0,1,0,0)
F2=(0,0,1,1,0,1,1,1,1,1,0,0,1,1,0,0)
F3=(0,1,0,1,1,0,0,0,0,1,0,1,1,0,1,0)
F4=(1,1,0,0,1,0,1,1,1,1,0,0,0,0,1,1)

Вариант 3

F1=(1,1,1,0,0,0,1,1,0,0,1,1,1,1,0,0)
F2=(0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,1)
F3=(0,1,0,0,1,0,1,0,0,1,0,1,1,0,1,0)
F4=(0,1,0,1,1,0,0,0,1,0,1,0,0,1,0,1)

Вариант 4

F1=(0,0,1,1,1,1,0,0,1,1,0,0,0,0,1,0)
F2=(0,1,1,0,0,0,1,0,1,0,0,1,1,0,0,1)
F3=(1,1,1,1,0,0,0,0,0,0,0,0,0,0,1,1,1)
F4=(0,1,1,0,0,1,1,0,1,1,1,0,0,1,1,0)

Вариант 5

F1=(0,1,0,1,1,0,1,0,0,1,0,1,1,0,1,1)

F2=(0,1,1,0,1,0,0,1,0,1,1,1,1,0,0,1)
F3=(1,1,1,0,1,1,0,0,0,0,1,1,0,0,1,1)
F4=(0,0,0,0,1,1,1,1,0,1,1,1,0,0,0,0)

Вариант 6

F1=(0,1,0,1,1,0,1,0,0,0,1,0,0,1,0,1)
F2=(0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,1)
F3=(1,1,0,0,0,0,1,1,1,1,0,0,0,1,1,1)
F4=(0,0,0,1,1,1,0,0,1,1,0,0,0,0,1,1)

Вариант 7

F1=(0,1,1,0,1,0,0,1,0,1,1,0,1,0,0,0)
F2=(0,0,1,1,0,0,1,1,1,1,0,0,1,0,0,0)
F3=(1,0,1,0,0,1,0,1,1,1,1,0,0,1,0,1)
F4=(0,1,1,0,0,1,0,0,1,0,0,1,1,0,0,1)

Вариант 8

F1=(0,1,0,1,1,0,1,0,1,0,1,0,0,1,0,0)
F2=(0,1,1,0,0,1,1,0,1,1,1,0,0,1,1,0)
F3=(0,1,1,1,1,0,0,1,0,1,1,0,1,0,0,1)
F4=(1,0,0,0,0,1,0,1,1,0,1,0,0,1,0,1)

Вариант 9

F1=(0,1,1,0,0,1,1,0,1,0,0,1,1,0,0,0)
F2=(1,1,0,0,0,0,1,0,0,0,1,1,1,1,0,0)
F3=(0,1,1,0,0,1,0,0,0,1,1,0,0,1,1,0)
F4=(0,0,0,0,1,1,1,1,1,1,1,1,0,1,0,0)

Вариант 10

F1=(1,0,1,0,0,0,0,1,1,0,1,0,0,1,0,1)
F2=(0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1)

F3=(0,1,0,1,1,0,1,0,1,0,0,0,0,1,0,1)
F4=(0,1,1,0,0,1,1,0,1,0,0,0,1,0,0,1)

Вариант 11

F1=(1,1,0,1,0,1,1,0,1,0,0,1,0,1,1,0)
F2=(0,1,0,1,0,1,0,1,0,0,1,0,1,0,1,0)
F3=(0,1,1,0,0,1,1,0,1,0,0,0,1,0,0,1)
F4=(0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,1)

Вариант 12

F1=(0,0,1,1,1,1,0,0,1,1,0,0,0,0,1,0)
F2=(0,1,0,1,0,1,0,1,1,0,1,0,0,0,1,0)
F3=(0,0,1,1,1,1,0,0,0,0,1,1,1,0,0,0)
F4=(1,0,0,0,0,1,0,1,1,0,1,0,0,1,0,1)

Вариант 13

F1=(0,0,1,1,1,1,0,0,0,0,1,1,1,1,0,1)
F2=(0,1,0,1,0,1,0,1,1,0,1,0,1,0,0,0)
F3=(1,1,0,1,0,0,0,0,0,0,0,0,1,1,1,1)
F4=(0,0,1,1,1,0,0,0,1,1,0,0,0,0,1,1)

Вариант 14

F1=(1,0,1,0,0,1,0,1,0,1,0,1,0,0,1,0)
F2=(0,1,0,1,0,1,0,0,1,0,1,0,1,0,1,0)
F3=(0,1,1,0,1,0,0,1,0,1,1,0,1,0,0,0)
F4=(0,1,0,1,1,0,1,0,0,1,0,1,1,1,1,0)

Вариант 15

F1=(0,0,1,1,0,0,1,1,1,0,0,0,1,1,0,0)
F2=(0,1,1,0,0,1,1,0,1,0,0,1,1,0,0,0)
F3=(1,1,0,0,0,0,1,1,0,0,1,1,1,0,0,0)
F4=(0,1,1,1,0,1,1,0,0,1,1,0,0,1,1,0)

Вариант 16

F1=(0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0)

F2=(1,0,1,0,0,1,0,1,0,1,0,1,1,0,0,0)

F3=(0,1,1,0,0,1,1,0,1,1,1,0,0,1,1,0)

F4=(0,1,1,0,1,0,0,1,0,1,1,0,1,0,0,0)

Вариант 17

F1=(0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,1)

F2=(0,1,0,1,0,0,1,0,1,0,1,0,0,1,0,1)

F3=(0,1,0,1,0,1,0,1,1,0,1,0,1,0,0,0)

F4=(1,0,0,1,1,1,0,1,1,0,0,1,1,0,0,1)

Вариант 18

F1=(0,1,0,1,1,0,1,0,1,0,1,0,0,0,0,1)

F2=(0,0,0,0,1,1,1,1,1,1,1,1,0,0,0,1)

F3=(0,1,0,1,0,1,0,1,1,0,1,0,1,0,0,0)

F4=(1,1,0,0,0,0,1,1,0,1,1,1,1,1,0,0)

Вариант 19

F1=(1,0,0,1,1,0,0,1,0,0,0,1,1,0,0,1)

F2=(0,0,0,1,0,1,0,1,1,0,1,0,1,0,1,0)

F3=(0,1,0,1,1,0,1,0,1,1,1,0,0,1,0,1)

F4=(0,1,1,0,0,1,1,0,1,0,0,1,1,0,0,0)

Вариант 20

F1=(0,1,0,1,0,1,0,1,1,0,1,0,1,0,1,1)

F2=(0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0)

F3=(1,1,0,0,1,1,0,0,0,0,1,1,0,0,0,1)

F4=(0,1,1,0,0,1,1,0,1,1,0,1,1,0,0,1)

ЛАБОРАТОРНАЯ РАБОТА 3. ВЗЛОМ 3-Х РАУНДОВОЙ СИММЕТРИЧНОЙ СИСТЕМЫ ШИФРОВАНИЯ ТИПА DES

Рассмотрим систему шифрования, построенную по схеме Фейстеля, которая использует слабый s-бокс предыдущего задания. Продемонстрируем, как можно взломать эту систему и найти секретный ключ, используя частичные зашифрованные данные (входы и выходы), полученные после третьего раунда процедуры шифрования.

3.1. Описание системы шифрования

На входе подается 2n-битовый вектор X, который разбивается на левую и правые половины, обозначаемые L и R. Левая половина подвергается преобразованию с помощью некоторой функции $f = f(K, L)$ (функции Фейстеля), где K – секретный ключ.

Результат побитно складывается с правой половиной R и отправляется на левую половину следующего раунда. Также левая половина 1-раунда без изменения передается в правую половину следующего раунда (рис. 3).

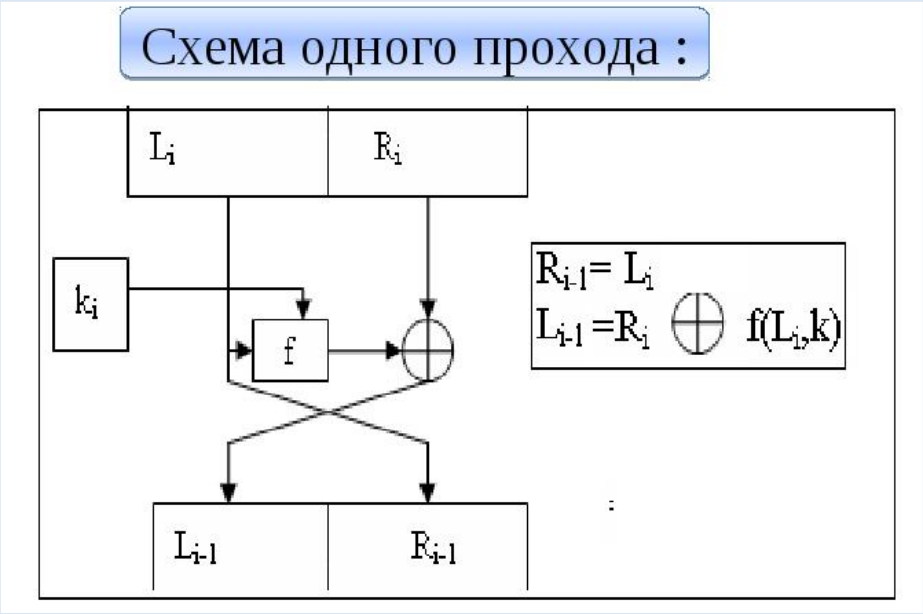


Рис. 3. Один проход системы Фейстеля

В каждом последующем раунде все операции повторяются. В оригинальной системе Фейстеля ключ K меняется от раунда к раунду и представляет собой выборку из исходного 64-битового ключа.

Опишем теперь (упрощенный) ***механизм функции f.*** На ее вход подается *n*-битовый вектор *L* и *n*-битовый ключ K (рис. 4).

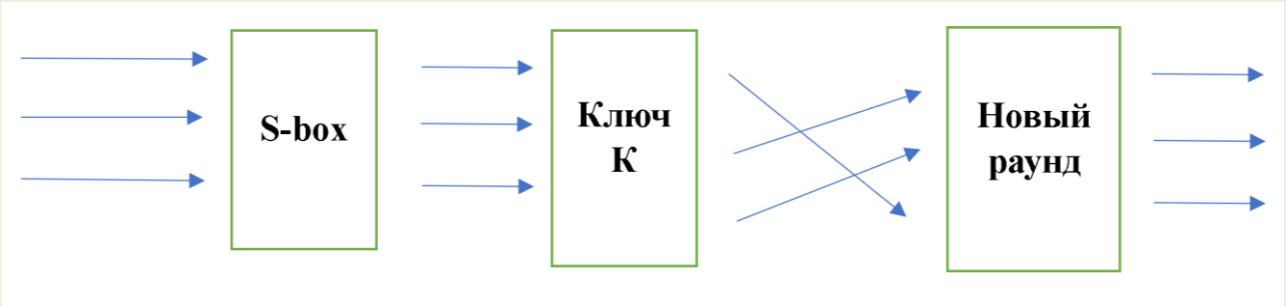


Рис. 4. Схема упрощенной функции Фейстеля f для нашей системы

Сначала исходный вектор *L* подвергается преобразованию с помощью n-мерного s-бокса. Потом результат побитно складывается с ключом K и подвергается перестановке P. Результатом будем такой же *n*-мерный вектор, который встраивается для дальнейшего преобразования в схему Фейстеля.

Отметим, что полный перебор системы шифрования требует перебора всех *2n*-битовых входов и n-битовых ключей, поэтому пространство перебора составляет величину 2^{3n}, что достаточно трудоемко для больших n ($n \geq 32$).

Покажем, как можно взломать данную систему, используя слабую нелинейность s-бокса. Будем предполагать, что в нашем распоряжении имеется сама система шифрования, позволяющая снимать выходные данные $(u_1, u_2, \dots u_n)$ при подаче на вход произвольного входа $(x_1, x_2, \dots, x_n)$.

Будем рассматривать пример размерности *n*=4. Пусть s-бокс задан следующими базисными функциями:

F1=(1,1,1,1,0,1,0,0,1,1,1,1,0,0,0,0)

F2=(0,0,0,1,0,1,0,1,0,1,0,1,0,1,0,1)

F3=(0,1,1,0,1,0,0,1,0,1,1,0,1,0,0,0)

F4=(0,1,0,1,0,1,0,1,1,0,0,0,1,0,1,0)

Пусть перестановка P представляет собой циклический сдвиг вправо $(0,1,2,3,) \rightarrow (1,2,3,0)$. Ключ $K = (k_0, k_1, k_2, k_3)$ является неизвестным.

Решение.

Составим сначала таблицу линейной аппроксимации T:

X 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

---

0 [8, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
1 [1, 1, -1, -1, 1, 1, -1, -1, -1, **7**, 1, 1, -1, -1, 1, 1] y3
2 [1, -1, -1, 1, -1, 1, 1, **7**, -1, 1, 1, -1, 1, -1, -1, 1] y2
3 [0, -2, 0, 2, -2, 0, 2, 0, 0, 2, 0, -2, 2, 0, 6, 0]
4 [1, **7**, 1, -1, 1, -1, 1, -1, 1, -1, 1, -1, 1, -1, 1, -1] y1
5 [0, 2, -2, 0, 0, 2, -2, 0, 6, 0, 0, 2, -2, 0, 0, 2]
6 [-2, 2, 0, 0, 0, 0, 6, 2, 0, 0, -2, 2, -2, 2, 0, 0]
7 [-1, -1, 3, -1, 1, -3, 1, 1, 3, -1, -1, -1, 1, 1, 1, 5]
8 [-1, 1, -1, 1, -**7**, -1, 1, -1, -1, 1, -1, 1, 1, -1, 1, -1] y0
9 [0, -2, 2, 0, -2, 0, 0, 2, 2, 0, 0, -2, 0, -6, -2, 0]
0 [0, 0, -2, -6, 0, 0, 2, -2, -2, 2, 0, 0, 2, -2, 0, 0]
1 [3, -1, -1, -1, -1, 3, -1, -1, -1, -1, -5, -1, -1, -1, -1,3]
2 [0, 0, 0, 0, -2, -6, -2, 2, 0, 0, 0, 0, -2, 2, -2, 2]
3 [-1, -1, 1, 1, 1, -3, 3, -1, 1, 1, -1, -1, -5, -1, 1,-3]
4 [1, -1, -5, -3, 1, -1, -1, 1, 3, -3, 1, -1, -1, 1, 1,-1]
5 [-2, 4, -2, 0, 2, 0, -2, 0, -2, 0, -2, -4, -2, 0, 2, 0]

Соответствующая система линейных уравнений имеет вид:

$$\begin{cases} Y_0 = X_1 + 1 \\ Y_1 = X_3 \\ Y_2 = X_1 + X_2 + X_3 \\ Y_3 = X_0 + X_3 \, . \end{cases} \quad (3.1)$$

Ключ шифрования нам неизвестен, но известна перестановка P, которая в нашем примере представляет собой циклический сдвиг влево на 1 позицию. Такую перестановку можно записать в виде (таблица 11), или в сокращенном виде $P = [3, 0, 1, 2]$.

Таблица 11

Перестановка

0	1	2	3
3	0	1	2

Зная эти данные, мы можем создать тестовую систему, заменяя s-бокс системой линейных уравнений и подставляя в качестве ключа K произвольные комбинации битов. Наша задача – определить ключ K, сравнивая выходы оригинальной системы шифрования и нашей тестовой системы.

3.2. Поиск 4-битного ключа 3-раундовой системы Фейстеля

Выполним отслеживание изменений 8-мерного входного кортежа (n=4): $X = (x_0, x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ в процессе шифрования по схеме Фейстеля. Обозначим через W выходной вектор, полученный на третьем раунде. Нас будет интересовать только правая часть W, которая согласно схеме Фейстеля совпадает с выходом левой части предыдущего раунда. Поэтому нам достаточно отследить преобразования двух раундов и получить левую половину выходного вектора второго раунда.

В соответствии со схемой Фейстеля исходный вектор X разбивается на левую и правую части L и R.

1. Левая часть $L = (x_0, x_1, x_2, x_3)$ подвергается преобразованию с помощью s-бокса, который дает выход $S(L) = (y_0, y_1, y_2, y_3)$.

2. Вектор $S(L) = (y_0, y_1, y_2, y_3)$ складывается побитно с ключом K:

$$Z = (y_0 + k_0, y_1 + k_1\ , y_2 + k_2, y_3 + k_3).$$

3. Вектор Z подвергается перестановке P:

$$U(u_0, u_1, u_2, u_3) = P(Z).$$

4. Вектор U складывается с правой половиной вектора X и подается на левую половину входа второго раунда:

$$L_2 = U + R = (u_0 + x_4, u_1 + x_5, u_2 + x_6, u_3 + x_7).$$

5. Полный вход второго раунда имеет вид:

$$X' = (u_0 + x_4, u_1 + x_5, u_2 + x_6, u_3 + x_7, x_0, x_1, x_2, x_3).$$

6. Вектор L_2 подвергается тем же преобразованиям с помощью операций пунктов 2–5 и выдается на выход системы в виде левой половины $2n$-мерного выходного вектора.

Последний вектор совпадает с правой половиной выходного вектора W полной 3-раундовой системы шифрования, частичные значения которой нам известны. Пользуясь слабостью s-бокса, мы можем построить тестовую систему шифрования, заменяя s-бокс системой линейных уравнений (3.1).

3.3. Тестовая система шифрования

Тестовая система имеет те же преобразования, что и основная система за исключением того, что s-бокс заменяется системой (3.1), то есть, выполняются уравнения (3.1), сложение с ключом K, перестановка $P = [3, 0, 1, 2]$ (циклический сдвиг влево на 1 позицию) и добавление правой части R. Перепишем систему (3.1):

$$\begin{cases} \boldsymbol{Y_0 = X_1 + 1} \\ \boldsymbol{Y_1 = X_3} \\ \boldsymbol{Y_2 = X_1 + X_2 + X_3} \\ \boldsymbol{Y_3 = X_0 + X_3}\,. \end{cases}$$

$$X(x_0, x_1, x_2, x_3) \to Y(y_0, y_1, y_2, y_3),$$

$$Y = ((x_1 + 1), (x_3), (x_1 + x_2 + x_3), (x_0 + x_3))\,,$$

$$Y + K = ((x_1 + 1 + k_0), (x_3 + k_1), (x_1 + x_2 + x_3 + k_2), (x_0 + x_3 + k_3))\ ,$$

$$P(Y + K) = ((x_0 + x_3 + k_3), (\,x_1 + 1 + k_0), (x_3 + k_1), (x_1 + x_2 + x_3 + k_2))\,,$$

$$W_1 = ((x_0 + x_3 + x_4 + k_3), (\,x_1 + x_5 + 1 + k_0), (x_3 + x_6 + k_1),$$

$$(x_1 + x_2 + x_3 + x_7 + k_2)).$$

Последний вектор $W_1(w_0, w_1, w_2, w_3)$ был получен сложением выхода 1-раунда с правой половиной $R(x_4, x_5, x_6, x_7)$ исходного век-

тора X. Далее он подается на место левой половины входного вектора второго раунда, где правая половина – это исходный вектор x_0, x_1, x_2, x_3 и вычисления повторяются. Итак, входной вектор второго раунда имеет вид:

$$X' = (w_0, w_1, w_2, w_3, x_0, x_1, x_2, x_3).$$

Далее, применяет к вектору W_1 функцию Фейстеля. После применения системы уравнения (3.1) получим вектор $Y'({y_0}', {y_1}', {y_2}', {y_3}')$ со следующими координатами:

$$\begin{cases} y_0' = w_1 + 1 = x_1 + x_5 + k_0 \\ y_1' = w_3 = x_1 + x_2 + x_3 + x_7 + k_2 \\ y_2' = w_1 + w_2 + w_3 = x_2 + x_5 + x_6 + x_7 + k_0 + k_1 + k_2 + 1 \\ y_3' = w_0 + w_3 = x_0 + x_1 + x_2 + x_4 + x_7 + k_2 + k_3 \,. \end{cases}$$

Прибавляем ключ K:

$$\begin{cases} z_0' = w_1 + 1 = x_1 + x_5 \\ z_1' = w_3 = x_1 + x_2 + x_3 + x_7 + k_2 + k_1 \\ z_2' = w_1 + w_2 + w_3 = x_2 + x_5 + x_6 + x_7 + k_0 + k_1 + 1 \\ z_3' = w_0 + w_3 = x_0 + x_1 + x_2 + x_4 + x_7 + k_2. \end{cases}$$

После выполнения перестановки получим вектор W_2 со следующими координатами:

$$\begin{cases} w_0' = x_0 + x_1 + x_2 + x_4 + x_7 + k_2 \\ {w_1}' = x_1 + x_5 \\ w_2' = x_1 + x_2 + x_3 + x_7 + k_1 + k_2 \\ {w_3}' = x_2 + x_5 + x_6 + x_7 + k_0 + k_1 + 1. \end{cases}$$

Последний шаг – это сложение вектора W_2 с правой половиной входного вектора второго раунда, то есть с вектором (x_0, x_1, x_2, x_3). В результате, получим вектор $U(u_0, u_1, u_2, u_3)$:

$$\begin{cases} u_0 = x_1 + x_2 + x_4 + x_7 + k_2 \\ u_1 = x_5 \\ u_2 = x_1 + x_3 + x_7 + k_1 + k_2 \\ u_3 = x_2 + x_5 + x_6 + x_7 + k_0 + k_1 + 1. \end{cases} \quad (3.2)$$

Отметим, что из-за плохой конструкции s-бокса мы получили достаточно простую аппроксимирующую систему, а второе уравнение системы вообще не зависит от ключа.

3.4. Поиск ключа

Сначала мы найдем значения аргументов X, на которых значения системы (3.1) отличаются от значения s-бокса. Для этого в цикле по всем аргументам X от (0,0,0,0) до (1,1,1,1) вычислим значения s-бокса и системы (3.1) и выделим значения, на которых они отличаются. Получим некоторый набор из 4-х значений. В десятичном формате этот набор имеет вид $\mathrm{Err} = \{5,1,15,10\}$. В нем i-уравнение системы (3.1) дает одну ошибку на i-аргументе множества Err. Например, $y_0(x) \neq F_0(x)$ на аргументе $x = 5 = 0101_2$.

Далее, подадим на вход нашей системы шифрования несколько аргументов, отличных от аргументов из набора Err и выпишем полученные результаты шифрования в таблицу. Будем брать значения правой половины входных векторов $R(X) = (x_4, x_5, x_6, x_7)$ равными 0. Значения левой половины будем брать последовательно от 0000 до 1000, исключая значения из множества Err. Полученные 7 значений выпишем в таблицу (для нашего примера мы взяли ключ шифрования $K = 0111_2 = 7$) (таблица 12).

Таблица 12

Таблица значения на аргументах X с правой частью 0 и ключом $K = 0111_2 = 7$

$L(X)$	0000	0010	0011	0100	0110	0111	1000
$\mathrm{C}(X)$	1001	0001	0110	1010	1011	1000	1000

Следующим этапом является вычисление значений тестовой системы (3.2) на аргументах из таблицы 12. Удалим из системы бесполезное второе уравнение и подставим 0 вместо переменных $x_4 - x_7$. Система (3.2) получит упрощенный вид:

$$\begin{cases} u_0 = x_1 + x_2 + k_2 \\ u_2 = x_1 + x_3 + k_1 + k_2 \\ u_3 = x_2 + k_0 + k_1 + 1. \end{cases} \quad (3.3)$$

Перепишем систему (3.3) в следующем виде:

$$\begin{cases} k_2 = u_0 + x_1 + x_2 \\ k_1 + k_2 = u_2 + x_1 + x_3 \\ k_0 + k_1 = u_3 + x_2 + 1. \end{cases} \quad (3.4)$$

Вычислим значения вектора:

$$Z = (u_0 + x_1 + x_2,\ u_2 + x_1 + x_3,\ u_3 + x_2 + 1)$$

на аргументах X из таблицы 12, где вместо переменных $U = u_0, u_1, u_2, u_3$ будем подставлять значения $C(X)$ из той же таблицы. Получим (таблица 13):

Таблица 13

Таблица перехваченных значений

$L(X)$	0000	0010	0010	0100	0110
$Z(X, C)$	101	101	101	101	101

Нам повезло, и система (3.4) имеет однозначное решение (1,0,1). Если бы не все значения были одинаковыми, то мы бы выбирали для каждой координаты то значение, которое появляется чаще. Выпишем решение системы (3.4) и найдем значения бит ключа

$$\begin{cases} k_2 = 1 \\ k_1 + k_2 = 0 \\ k_0 + k_1 = 1 \end{cases} \quad \rightarrow \quad \begin{cases} k_0 = 0 \\ k_1 = 1 \\ k_2 = 1. \end{cases}$$

Последний бит ключа k_3 нам не нужен, поскольку он не используется в системе. Взлом ключа выполнен!

3.5. Задание на лабораторную работу 3

Для своего варианта взять данные s-бокса из лабораторной работы 2 и выполнить следующее:

1. Построить систему линейных уравнений (3.2).
2. Выполнить поиск неизвестного ключа, построив систему (3.4).
3. Зная ключ, расшифровать перехваченное сообщение c(m).

Варианты заданий

Вариант 1

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(6,6,1,15,13,15,6,6,11,2).
Перестановка P=[2,1,3,0].
Зашифрованное сообщение c(m)=8.

Вариант 2

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(7,14,8,0,11,15,6,10,8,8).
Перестановка P=[3,0,1,2].
Зашифрованное сообщение c(m)=5, 9<=m<16.

Вариант 3

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(15,4,13,0,13,14,2,6,4,3).
Перестановка P=[1,3,0,2].
Зашифрованное сообщение c(m)=3, 9<=m<16.

Вариант 4

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(10,13,2,7,12,11,7,2,5,12).
Перестановка P=[3,1,2,0].
Зашифрованное сообщение c(m)=13, 9<=m<16.

Вариант 5

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(0,9,15,0,6,11,7,6,14,7).
Перестановка P=[3,1,2,0].
Зашифрованное сообщение c(m)=9, 9<=m<16.

Вариант 6

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(12,8,5,3,13,15,4,4,13,11).

Перестановка P=[1,2,3,0].

Зашифрованное сообщение c(m)=7, 9<=m<16.

Вариант 7

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(13,5,13,4,10,11,1,2,0,1).

Перестановка P=[2,3,0,1].

Зашифрованное сообщение c(m)=0, 9<=m<16.

Вариант 8

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(6,10,6,4,3,3,10,14,13,6).

Перестановка P=[1,0,3,2].

Зашифрованное сообщение c(m)=8, 9<=m<16.

Вариант 9

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(7,14,0,9,2,9,14,1,13,5).

Перестановка P=[1,3,2,0].

Зашифрованное сообщение c(m)=5, 9<=m<16.

Вариант 10

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(12,4,1,12,2,15,11,1,8,15).

Перестановка P=[3,0,1,2].

Зашифрованное сообщение c(m)=12, 9<=m<16.

Вариант 11
Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(4,8,7,5,11,9,8,10,6,10).
Перестановка P=[3,2,0,1].
Зашифрованное сообщение c(m)=4, 9<=m<16.

Вариант 12
Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(10,7,0,8,6,11,9,4,1,5).
Перестановка P=[1,2,3,0].
Зашифрованное сообщение c(m)=12, 9<=m<16.

Вариант 13
Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(9,2,9,7,15,0,10,0,0,0).
Перестановка P=[3,1,0,2].
Зашифрованное сообщение c(m)=8, 9<=m<16.

Вариант 14
Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(13,9,0,4,15,9,0,5,13,15).
Перестановка P=[3,0,1,2].
Зашифрованное сообщение c(m)=4, 9<=m<16.

Вариант 15
Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:
C=(4,0,8,2,11,1,3,11,0,0).
Перестановка P=[1,3,2,0].
Зашифрованное сообщение c(m)=0, 9<=m<16.

Вариант 16

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(0,8,1,8,13,13,4,12,10,4).

Перестановка P=[1,3,0,2].

Зашифрованное сообщение c(m)=2, 9<=m<16.

Вариант 17

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(3,5,12,0,6,9,14,11,2,15).

Перестановка P=[2,1,0,3].

Зашифрованное сообщение c(m)=1, 9<=m<16.

Вариант 18

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(1,11,12,14,14,6,13,3,15,5).

Перестановка P=[2,3,0,1].

Зашифрованное сообщение c(m)=2, 9<=m<16.

Вариант 19

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(7,14,6,14,12,5,9,1,2,7).

Перестановка P=[2,0,1,3].

Зашифрованное сообщение c(m)=7, 9<=m<16.

Вариант 20

Частичная шифровальная таблица 3-раундового шифра Фейстеля от X=0 до X=8:

C=(2,9,12,7,2,9,8,5,3,7).

Перестановка P=[2,1,0,3].

Зашифрованное сообщение c(m)=7, 9<=m<16.

ЛИТЕРАТУРА

1. A new S-box pattern generation based on chaotic enhanced logistic map. – URL: https://cybersecurity.springeropen.com/articles/10.1186/s42400-024-00254-4 (дата обращения: 12.08.2025).

2. Howard M. Heys. A Tutorial on Linear and Differential Cryptanalysis. – URL: http://www.engr.mun.ca/~howard/PAPERS/ldc_tutorial.pdf (дата обращения: 03.09.2025).

3. Дифференциальный криптоанализ для чайников. – URL: https://habr.com/ru/articles/215527/ (дата обращения: 03.09.2025).

Учебное издание

Ишмухаметов Шамиль Талгатович,

Мубараков Булат Газинурович

ЭЛЕМЕНТЫ ЛИНЕЙНОГО КРИПТОАНАЛИЗА

Учебно-методическое пособие