Оглавление

Линейный криптоанализ

Метод линейного криптоанализа разработан в 1993 году японским криптологом Митсуру Матсуи (Mitsuru Matsui). В первоначальном виде этот метод сформулирован применительно к криптосистеме DES [1].

Рассмотрим алгоритм шифрования, построенный на основе сети SPN, структура которого показана на рис. 1. Здесь $X=(x_1,x_2,...,x_{16})$ - 16-ти битовый блок открытого (исходного) сообщения, $Y=(y_1,y_2,...,y_{16})$ - 16-ти битовый блок закрытого (зашифрованного) сообщения. В основе алгоритма – последовательное применение двух основных преобразований: замены π_S

$$\pi_S:\{0,1\}^l \to \{0,1\}^l$$

и перестановки π_P

$$\pi_P:\{1,...lm\} \to \{1,...,lm\},$$

где $lm \in Z$ - размер блока. В алгоритме, представленном на рис. 1, l =4, m =4.

Преобразование π_S можно задать в виде таблицы, где первая строка задает вход (z), а вторая строка – выход ($\pi_S(z)$). Табл.1 задает используемое в данном алгоритме преобразование π_S. На рис. 1 это преобразование показано в виде S-блоков замены.

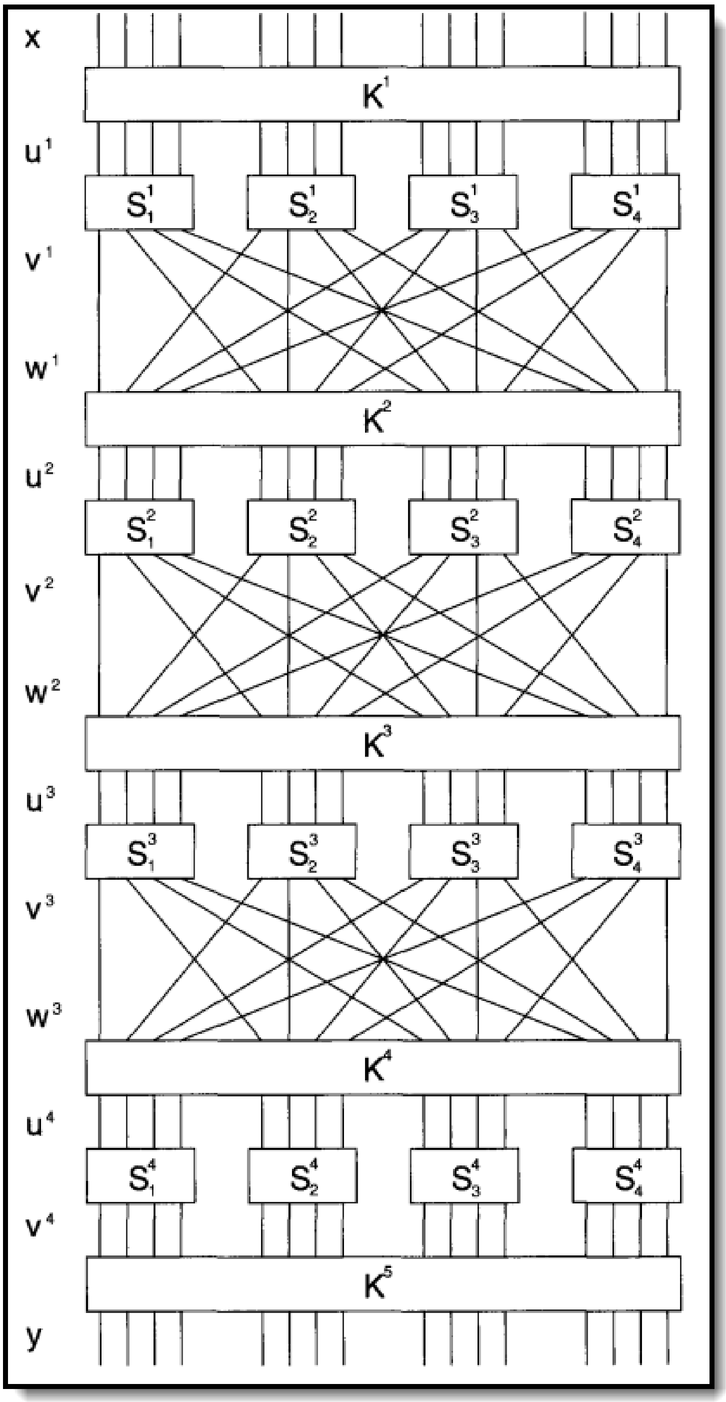

Рис. 1. Структура алгоритма шифрования, построенного на основе сети SPN [3]

Для выполнения замены π_S необходимо выполнить следующие шаги:

1. lm-битовый блок разбить на m l-битовых подблоков. Такое разбиение для lm-битового блока $u=(u_1,...,u_{lm})$ можно записать таким образом

$$u = u_{\langle 1\rangle} \| ... \| u_{\langle m\rangle},$$

где $u_{\langle i\rangle} = \left(u_{(i-1)\cdot l+1},...,u_{i\cdot l}\right),\ i=1,...,m$

2. Применить преобразование π_S над каждым подблоком:

$$v_{\langle i\rangle} = \pi_S(u_{\langle i\rangle}),\ i=1,...,m$$

3. Объединить подблоки в один lm-битовый вход

$$v = v_{\langle 1\rangle} \| ... \| v_{\langle m\rangle}.$$

Пример. Путь $l=4, m=4$. Тогда 16-ти битовый блок $u=(u_1,...,u_{16})$ разбивается на 4 подблока $u = u_{\langle 1\rangle} \| ... \| u_{\langle 4\rangle}$, где

$$u_{\langle 1\rangle} = \left(u_1,...,u_4\right),$$

$$u_{\langle 2\rangle} = \left(u_5,...,u_8\right),$$

$$u_{\langle 3\rangle} = \left(u_9,...,u_{12}\right),$$

$$u_{\langle 4\rangle} = \left(u_{13},...,u_{16}\right).$$

Для блока $u=(0,1,1,0,1,1,0,1,0,1,1,1,0,0,1,0)$ получим 4 подблока:

$$u_{\langle 1\rangle} = \left(0,1,1,0\right).$$

$$u_{\langle 2\rangle} = \left(1,1,0,0\right),$$

$$u_{\langle 3\rangle} = \left(0,1,1,1\right),$$

$$u_{\langle 4\rangle} = \left(0,0,1,0\right).$$

После применения преобразования π_S (табл.1) над каждым из подблоков получим

$$v_{\langle 1\rangle} = \left(1,0,1,1\right),$$

$$v_{\langle 2\rangle} = (0,1,0,1),$$

$$v_{\langle 3\rangle} = (1,0,0,0),$$

$$v_{\langle 4\rangle} = (1,1,0,1).$$

Таким образом, результатом применения преобразования замены π_S над блоком u будет блок $v = (1,0,1,1,0,1,0,1,1,0,0,0,1,1,0,1)$.

Таблица 1

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

Преобразование π_P задает перестановку бит внутри блока. Данное преобразование удобно задать в виде таблицы, в которой в первой строке (вход z) заданы порядковые номера i бит блока (нумерация слева-направо), а во второй строке - выход ($\pi_P(z)$) – результат перестановки бит внутри блока, т.е. на i-ю позицию ставится $\pi_P(i)$ бит блока.

Таблица 2

вход z	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
выход $\pi_P(z)$	0	4	8	12	1	5	9	13	2	6	10	14	3	7	11	15

Пример. Для блока $v = (1,0,1,1,1,1,0,1,0,0,0,1,1,1,0,1)$ результатом применения преобразования π_P (табл.2) будет блок $w = \pi_P(v)$:

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

Псевдокод алгоритма приведен на рис. 2. Составной частью алгоритма является описание процедуры получения раундовых ключей – так называемой процедуры генерации подключей. Для рассматриваемого алгоритма процедура генерации подключей заключается в следующем: все пять подключей получаются последовательным выбором 16 бит из 32 битного ключа $K=(k_1,...,k_{32})\in\{0,1\}^{32}$ по следующему правилу. Ключ K^r $(1\le r\le 5)$ состоит из 16 последовательных бит ключа K, начиная с $k_{4\cdot r-3}$.

$$
\begin{array}{l}
\mathrm{SPN}(x, \pi_S, \pi_P, (K^1, \ldots, K^{\mathrm{Nr}+1}))\\
w^0 \leftarrow x\\
\textbf{for } r \leftarrow 1 \textbf{ to } \mathrm{Nr}-1\\
\quad \textbf{do} \begin{cases} u^r \leftarrow w^{r-1} \oplus K^r\\ \textbf{for } i \leftarrow 1 \textbf{ to } m\\ \quad \textbf{do } v^r_{<i>} \leftarrow \pi_S(u^r_{<i>})\\ w^r \leftarrow (v^r_{\pi_P(1)}, \ldots, v^r_{\pi_P(\ell m)}) \end{cases}\\
u^{\mathrm{Nr}} \leftarrow w^{\mathrm{Nr}-1} \oplus K^{\mathrm{Nr}}\\
\textbf{for } i \leftarrow 1 \textbf{ to } m\\
\quad \textbf{do } v^{\mathrm{Nr}}_{<i>} \leftarrow \pi_S(u^{\mathrm{Nr}}_{<i>})\\
y \leftarrow v^{\mathrm{Nr}} \oplus K^{\mathrm{Nr}+1}\\
\textbf{output } (y)
\end{array}
$$

Рис.2. Псевдокод алгоритма, основанного на архитектуре сети SPN [3]

Пример. Для ключа

$$K = 0011\ 1010\ 1001\ 0100\ 1101\ 0110\ 0011\ 1111$$

в результате применения процедуры генерации подключей (расширения ключа) получим следующие раундовые ключи:

$$K^1 = 0011\ 1010\ 1001\ 0100$$
$$K^2 = 1010\ 1001\ 0100\ 1101$$
$$K^3 = 1001\ 0100\ 1101\ 0110$$
$$K^4 = 0100\ 1101\ 0110\ 0011$$
$$K^5 = 1101\ 0110\ 0011\ 1111.$$

Пример. Для открытого 16-ти битового блока

$$x = 0010\ 0110\ 1011\ 0111$$

Последовательное применение алгоритма (рис. 2) дает следующие результаты:

$$w^0 = 0010\ 0110\ 1011\ 0111$$
$$K^1 = 0011\ 1010\ 1001\ 0100$$
$$u^1 = 0001\ 1100\ 0010\ 0011$$
$$v^1 = 0100\ 0101\ 1101\ 0001$$
$$w^1 = 0010\ 1110\ 0000\ 0111$$
$$K^2 = 1010\ 1001\ 0100\ 1101$$
$$u^2 = 1000\ 0111\ 0100\ 1010$$
$$v^2 = 0011\ 1000\ 0010\ 0110$$
$$w^2 = 0100\ 0001\ 1011\ 1000$$
$$K^3 = 1001\ 0100\ 1101\ 0110$$
$$u^3 = 1101\ 0101\ 0110\ 1110$$
$$v^3 = 1001\ 1111\ 1011\ 0000$$
$$w^3 = 1110\ 0100\ 0110\ 1110$$
$$K^4 = 0100\ 1101\ 0110\ 0011$$
$$u^4 = 1010\ 1001\ 0000\ 1101$$
$$v^4 = 0110\ 1010\ 1110\ 1001$$
$$K^5 = 1101\ 0110\ 0011\ 1111,$$
$$y = 1011\ 1100\ 1101\ 0110$$

Накопительная лемма (The Pilling-up lemma)

Пусть $X_1, X_2, \ldots$ независимые случайные величины, принимающие значения из множества $\{0,1\}$. Пусть $p_1, p_2, \ldots$ действительные числа, такие что $0 \le p_i \le 1, i = 1, 2, \ldots$. Тогда, если

$$P(X_i = 0) = p_i, \tag{1}$$

то

$$P(X_i = 1) = 1 - p_i \tag{2}$$

Если две случайные величины независимы, то для $i \ne j$ верно

$$\begin{aligned} P(X_i = 0, X_j = 0) &= p_i p_j \\ P(X_i = 0, X_j = 1) &= p_i (1 - p_j) \\ P(X_i = 1, X_j = 0) &= (1 - p_i) p_j \\ P(X_i = 1, X_j = 1) &= (1 - p_i)(1 - p_j) \end{aligned}$$

Для случайной величины $X_1 \oplus X_2$ верно

$$\begin{aligned} P(X_1 \oplus X_2 = 0) &= p_i p_j + (1 - p_i)(1 - p_j) \\ P(X_1 \oplus X_2 = 1) &= p_i (1 - p_j) + (1 - p_i) p_j \end{aligned}$$

Пусть $\varepsilon_1, \varepsilon_2, \ldots$ действительные числа, такие что $-\frac{1}{2} \le \varepsilon_i \le \frac{1}{2}, i = 1, 2, \ldots$

Тогда, ε_i задает отклонение случайной величины p_i от $\frac{1}{2}$:

$$\varepsilon_i = p_i - \frac{1}{2}.$$

В этом случае (1, 2) можно записать в виде

$$P(X_i = 0) = \frac{1}{2} + \varepsilon_i$$

$$P(X_i = 1) = \frac{1}{2} - \varepsilon_i.$$

Узнать отклонение для случайной величины $X_{i_1} \oplus \cdots \oplus X_{i_k}$ позволяет накопительная лемма.

Накопительная лемма. Пусть $\varepsilon_{i_1,i_2,\ldots,i_k}$ обозначает отклонение случайной величины $X_{i_1} \oplus \cdots \oplus X_{i_k}$. Тогда

$$\varepsilon_{i_1,i_2,\ldots,i_k} = 2^{k-1} \prod_{j=1}^{k} \varepsilon_{i_j} \, .$$

Линейное приближение S-блоков

Идея метода линейного криптоанализа основана на том, что существует возможность заменить нелинейную функцию криптографического преобразования ее линейным аналогом. Линейный криптоанализ базируется на знании криптоаналитиком пар «открытый текст-криптограмма», а также алгоритма шифрования.

В блочных алгоритмах, построенных на основе сети SPN, нелинейной операцией является операция замены (Sbox).

Таким образом, нелинейную операцию замены можно приблизить линейным выражением

$$\left(\bigoplus_{i=1}^{n} a_i X_i \right) = \left(\bigoplus_{i=1}^{n} b_i Y_i \right),$$

где $a_i \in \{0,1\}$, $b_i \in \{0,1\}$

Рассмотрим S-блок, изображенный на рис. 3 с входным вектором $X = (x_1,\ldots,x_4)$ и выходным вектором $Y = (y_1,\ldots y_4)$.

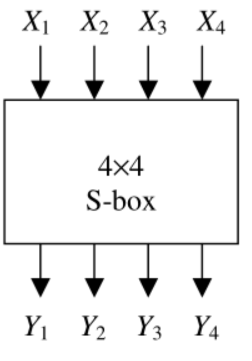

Рис. 3. S-блок

В соответствии с табл. 1 запишем результаты всех возможных замен для вектора X в табл.3.

Таблица 3

X_1	X_2	X_3	X_4	Y_1	Y_2	Y_3	Y_4
0	0	0	0	1	1	1	0
0	0	0	1	0	1	0	0
0	0	1	0	1	1	0	1
0	0	1	1	0	0	0	1
0	1	0	0	0	0	1	0
0	1	0	1	1	1	1	1
0	1	1	0	1	0	1	1
0	1	1	1	1	0	0	0
1	0	0	0	0	0	1	1
1	0	0	1	1	0	1	0
1	0	1	0	0	1	1	0
1	0	1	1	1	1	0	0
1	1	0	0	0	1	0	1
1	1	0	1	1	0	0	1
1	1	1	0	0	0	0	0
1	1	1	1	0	1	1	1

Рассмотрим случайную величину $X_1 \oplus X_4 \oplus Y_2$ в качестве линейного приближения данной таблицы замен. Определить вероятность $P(X_1 \oplus X_4 \oplus Y_2 = 0)$ можно, если подсчитать по табл. 3 сколько раз выполняется равенство $X_1 \oplus X_4 \oplus Y_2 = 0$ и результат поделить на 16 (рис. 4).

Нетрудно убедиться, что

$$P(X_1 \oplus X_4 \oplus Y_2 = 0) = \frac{1}{2}$$

и, следовательно,

$$P(X_1 \oplus X_4 \oplus Y_2 = 1) = \frac{1}{2}$$

Это означает, что отклонение для данной случайной величины равно нулю и в качестве линейного приближения его использовать не рекомендуется.

X_1	X_2	X_3	X_4	Y_1	Y_2	Y_3	Y_4	$X_1 \oplus X_4$	Y_2	$X_3 \oplus X_4$	$Y_1 \oplus Y_4$
0	0	0	0	1	1	1	0	0	1	0	1
0	0	0	1	0	1	0	0	1	1	1	0
0	0	1	0	1	1	0	1	0	1	1	0
0	0	1	1	0	0	0	1	1	0	0	1
0	1	0	0	0	0	1	0	0	0	0	0
0	1	0	1	1	1	1	1	1	1	1	0
0	1	1	0	1	0	1	1	0	0	1	0
0	1	1	1	1	0	0	0	1	0	0	1
1	0	0	0	0	0	1	1	1	0	0	1
1	0	0	1	1	0	1	0	0	0	1	1
1	0	1	0	0	1	1	0	1	1	1	0
1	0	1	1	1	1	0	0	0	1	0	1
1	1	0	0	0	1	0	1	1	1	0	1
1	1	0	1	1	0	0	1	0	0	1	0
1	1	1	0	0	0	0	0	1	0	1	0
1	1	1	1	0	1	1	1	0	1	0	1

Рис. 4.

Если рассмотреть случайную величину $X_3 \oplus X_4 \oplus Y_1 \oplus Y_4$, нетрудно убедиться, что отклонение для нее составит $-\frac{3}{8}$ (рис. 4).

Далее необходимо найти отклонения для всех возможных случайных величин, записанных в виде

$$\left(\bigoplus_{i=1}^{n} a_i X_i \right) = \left(\bigoplus_{i=1}^{n} b_i Y_i \right), \qquad (3)$$

где $a_i \in \{0,1\}$, $b_i \in \{0,1\}$.

Бинарный вектор (a_1, a_2, a_3, a_4) представим в 16-ой системе счисления в виде значений от 0 до F (в табл. 4 они названы как Input Sum). Также поступим и с вектором (b_1, b_2, b_3, b_4) (в табл. 4 полученные значения подписаны как Output Sum). Тогда, случайную величину можно описать в виде пары (a,b), где $a = (a_1, a_2, a_3, a_4)$, $b = (b_1, b_2, b_3, b_4)$.

Пример. Для случайной переменной $X_1 \oplus X_4 \oplus Y_2$ вектор (a_1, a_2, a_3, a_4) равен $(1,0,0,1)$, что соответствует 9 в шестнадцатеричной системе счисления. Вектор $(b_1, b_2, b_3, b_4) = (0,1,0,0)$, что соответствует 4 в шестнадцатеричной системе счисления. Тогда случайную величину $X_1 \oplus X_4 \oplus Y_2$ можно записать в виде (9.4).

Для всех возможных значений $(x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4)$, таких что $(x_1, x_2, x_3, x_4) = \pi_S(y_1, y_2, y_3, y_4)$, для каждой случайной величины (a,b) необходимо подсчитать сколько раз выполняется равенство

$$\left(\bigoplus_{i=1}^{n} a_i x_i\right) \oplus \left(\bigoplus_{i=1}^{n} b_i y_i\right) = 0$$

Найденное значение $N_L(a,b)$ используется для вычисления отклонения для случайной величины (a,b) по следующей формуле

$$\varepsilon_{(a,b)} = \frac{N_L(a,b) - 8}{16}.$$

Пример. Для случайной величины $X_1 \oplus X_4 \oplus Y_2$ или (9,4), что эквивалентно. найденное значение $N_L(a,b)$ =8. Значит отклонение $\varepsilon_{(9,4)} = 0$.

В табл. 4 приведены найденные значения в виде $N_L(a,b) - 8$. Найденная таким образом таблица называется таблицей линейных приближений.

Таблица 4

Input Sum \ Output Sum	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	+8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	–2	–2	0	0	–2	+6	+2	+2	0	0	+2	+2	0	0
2	0	0	–2	–2	0	0	–2	–2	0	0	+2	+2	0	0	–6	+2
3	0	0	0	0	0	0	0	0	+2	–6	–2	–2	+2	+2	–2	–2
4	0	+2	0	–2	–2	–4	–2	0	0	–2	0	+2	+2	–4	+2	0
5	0	–2	–2	0	–2	0	+4	+2	–2	0	–4	+2	0	–2	–2	0
6	0	+2	–2	+4	+2	0	0	+2	0	–2	+2	+4	–2	0	0	–2
7	0	–2	0	+2	+2	–4	+2	0	–2	0	+2	0	+4	+2	0	+2
8	0	0	0	0	0	0	0	0	–2	+2	+2	–2	+2	–2	–2	–6
9	0	0	–2	–2	0	0	–2	–2	–4	0	–2	+2	0	+4	+2	–2
A	0	+4	–2	+2	–4	0	+2	–2	+2	+2	0	0	+2	+2	0	0
B	0	+4	0	–4	+4	0	+4	0	0	0	0	0	0	0	0	0
C	0	–2	+4	–2	–2	0	+2	0	+2	0	+2	+4	0	+2	0	–2
D	0	+2	+2	0	–2	+4	0	+2	–4	–2	+2	0	+2	0	0	+2
E	0	+2	+2	0	–2	–4	0	+2	–2	0	0	–2	–4	+2	–2	0
F	0	–2	–4	–2	–2	0	+2	0	0	–2	+4	–2	–2	0	+2	0

Построение линейного приближения для шифра

На рис. 5 приведена структура используемого линейного приближения. Это одно из возможных решений.

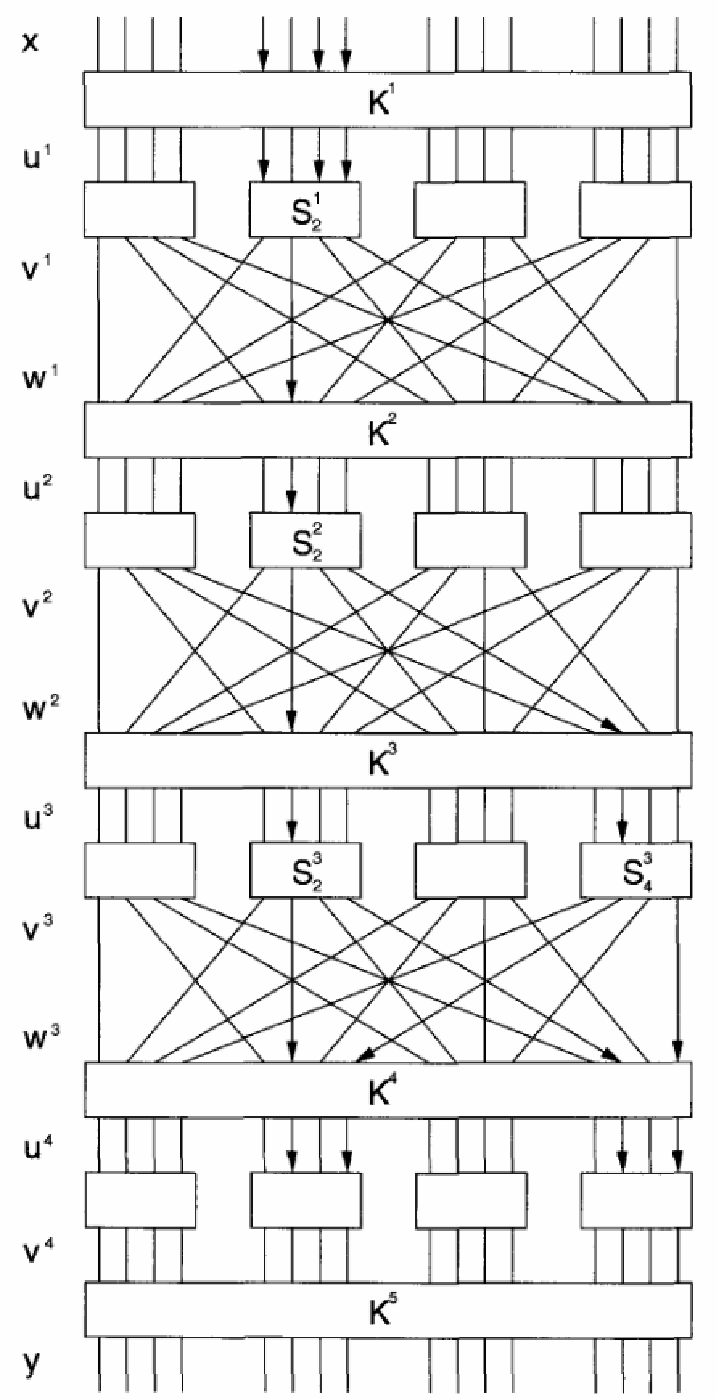

Рис. 5.

На рис. 5 линии со стрелками обозначают случайные величины, которые будут включены в линейное приближение.

Данное приближение включает линейные приближения следующих S-блоков.

$$\begin{aligned}
&S_2^1,\ \mathbf{T_1} = \mathbf{U_5^1} \oplus \mathbf{U_7^1} \oplus \mathbf{U_8^1} \oplus \mathbf{V_6^1}\\
&S_2^2,\ \mathbf{T_2} = \mathbf{U_6^2} \oplus \mathbf{V_6^2} \oplus \mathbf{V_8^2}\\
&S_2^3,\ \mathbf{T_3} = \mathbf{U_6^3} \oplus \mathbf{V_6^3} \oplus \mathbf{V_8^3}\\
&S_4^3,\ \mathbf{T_4} = \mathbf{U_{14}^3} \oplus \mathbf{V_{14}^3} \oplus \mathbf{V_{16}^3}
\end{aligned}$$

Случайные величины T_1, T_2, T_3, T_4 имеют отклонения, соответственно равные $\frac{1}{4}, -\frac{1}{4}, -\frac{1}{4}, -\frac{1}{4}$.

Применение накопительной леммы позволяет получить отклонение для случайной величины $T_1 \oplus T_2 \oplus T_3 \oplus T_4$:

$$2^3(1/4)(-1/4)^3 = -1/32.$$

Из рис. 5 следует, что

$$\begin{aligned}
\mathbf{T_1} &= \mathbf{U_5^1} \oplus \mathbf{U_7^1} \oplus \mathbf{U_8^1} \oplus \mathbf{V_6^1} &&= \mathbf{X_5} \oplus \mathbf{K_5^1} \oplus \mathbf{X_7} \oplus \mathbf{K_7^1} \oplus \mathbf{X_8} \oplus \mathbf{K_8^1} \oplus \mathbf{V_6^1}\\
\mathbf{T_2} &= \mathbf{U_6^2} \oplus \mathbf{V_6^2} \oplus \mathbf{V_8^2} &&= \mathbf{V_6^1} \oplus \mathbf{K_6^2} \oplus \mathbf{V_6^2} \oplus \mathbf{V_8^2}\\
\mathbf{T_3} &= \mathbf{U_6^3} \oplus \mathbf{V_6^3} \oplus \mathbf{V_8^3} &&= \mathbf{V_6^2} \oplus \mathbf{K_6^3} \oplus \mathbf{V_6^3} \oplus \mathbf{V_8^3}\\
\mathbf{T_4} &= \mathbf{U_{14}^3} \oplus \mathbf{V_{14}^3} \oplus \mathbf{V_{16}^3} &&= \mathbf{V_8^2} \oplus \mathbf{K_{14}^3} \oplus \mathbf{V_{14}^3} \oplus \mathbf{V_{16}^3}.
\end{aligned}$$

Следовательно, случайная величина

$$\begin{aligned}
&\mathbf{X_5} \oplus \mathbf{X_7} \oplus \mathbf{X_8} \oplus \mathbf{V_6^3} \oplus \mathbf{V_8^3} \oplus \mathbf{V_{14}^3} \oplus \mathbf{V_{16}^3}\\
&\qquad \oplus \mathbf{K_5^1} \oplus \mathbf{K_7^1} \oplus \mathbf{K_8^1} \oplus \mathbf{K_6^2} \oplus \mathbf{K_6^3} \oplus \mathbf{K_{14}^3}
\end{aligned}$$

имеет отклонение такое же, как и $T_1 \oplus T_2 \oplus T_3 \oplus T_4 (-\frac{1}{32})$.

Далее, так как (рис. 5)

$$\begin{aligned}
\mathbf{V_6^3} &= \mathbf{U_6^4} \oplus \mathbf{K_6^4}\\
\mathbf{V_8^3} &= \mathbf{U_{14}^4} \oplus \mathbf{K_{14}^4}\\
\mathbf{V_{14}^3} &= \mathbf{U_8^4} \oplus \mathbf{K_8^4}\\
\mathbf{V_{16}^3} &= \mathbf{U_{16}^4} \oplus \mathbf{K_{16}^4}
\end{aligned}$$

верно

$$\begin{aligned}
&\mathbf{X_5} \oplus \mathbf{X_7} \oplus \mathbf{X_8} \oplus \mathbf{U_6^4} \oplus \mathbf{U_8^4} \oplus \mathbf{U_{14}^4} \oplus \mathbf{U_{16}^4}\\
&\quad \oplus \mathbf{K_5^1} \oplus \mathbf{K_7^1} \oplus \mathbf{K_8^1} \oplus \mathbf{K_6^2} \oplus \mathbf{K_6^3} \oplus \mathbf{K_{14}^3} \oplus \mathbf{K_6^4} \oplus \mathbf{K_8^4} \oplus \mathbf{K_{14}^4} \oplus \mathbf{K_{16}^4}
\end{aligned} \tag{4}$$

Случайная величина (4) имеет отклонение $-\frac{1}{32}$. Запишем ее как сумму двух случайных величин: $\bar{X} \oplus \bar{K}$. Тогда, по накопительной лемме отклонение для $\bar{X} \oplus \bar{K}$ будет

$$\varepsilon_{XK} = 2\varepsilon_X \varepsilon_K = -\frac{1}{32}.$$

А так как, случайная величина $\bar{K}$ всегда равна 0 или 1, то отклонение у нее будет либо $\varepsilon_K = -\frac{1}{2}$, либо $\varepsilon_K = \frac{1}{2}$. Тогда,

$$\varepsilon_{XK} = 2\varepsilon_X \frac{1}{2} = \varepsilon_X = -\frac{1}{32}$$

или

$$\varepsilon_{XK} = 2\varepsilon_X \cdot -\frac{1}{2} = -\frac{1}{32}$$

$$\varepsilon_X = \frac{1}{32}$$

Т.е. случайная величина

$$\mathbf{X_5} \oplus \mathbf{X_7} \oplus \mathbf{X_8} \oplus \mathbf{U_6^4} \oplus \mathbf{U_8^4} \oplus \mathbf{U_{14}^4} \oplus \mathbf{U_{16}^4} \qquad (5)$$

имеет отклонение $\pm\frac{1}{32}$.

Предположим, что есть T пар открытый-зашифрованный текст. Пусть $\mathfrak{I}$ - множество таких пар. Проведение атаки с помощью рассмотренного приближения позволит получить 8 бит ключа K^5:

$$K_5^5, K_6^5, K_7^5, K_8^5, K_{13}^5, K_{14}^5, K_{15}^5,\ K_{16}^5.$$

Всего 256 вариантов (частичных ключей).

Для каждой пары $(x, y) \in \mathfrak{I}$ и каждого частичного ключа надо выполнить два шага расшифрования: наложить раундовый ключ K^5 с помощью операции «исключающее или» и выполнить обратную замену в блоках S_2^4 и S_4^4. В результате выполнения этих действий будут получены значения $u^4_{\langle 2\rangle}$ и $u^4_{\langle 4\rangle}$. Это позволит вычислить значение случайной величины (5):

$$x_5 \oplus x_7 \oplus x_8 \oplus u^4_6 \oplus u^4_8 \oplus u^4_{14} \oplus u^4_{16}. \qquad (6)$$

Для каждого значения частичного ключа значение счетчика увеличивается на 1, когда (6) равно 0. Счетчик, значение которого больше всего отличается от $\frac{T}{2}$ укажет на правильное значение частичного ключа.

Псевдокод алгоритма представлен на рис. 6.

```
LinearAttack(𝒯, T, π_S^{-1})

for (L1, L2) ← (0, 0) to (F, F)
  do Count[L1, L2] ← 0
for each (x, y) ∈ 𝒯
  do { for (L1, L2) ← (0, 0) to (F, F)
         do { v^4_<2> ← L1 ⊕ y_<2>
              v^4_<4> ← L2 ⊕ y_<4>
              u^4_<2> ← π_S^{-1}(v^4_<2>)
              u^4_<4> ← π_S^{-1}(v^4_<4>)
              z ← x5 ⊕ x7 ⊕ x8 ⊕ u^4_6 ⊕ u^4_8 ⊕ u^4_14 ⊕ u^4_16
              if z = 0
                then Count[L1, L2] ← Count[L1, L2] + 1
max ← −1
for (L1, L2) ← (0, 0) to (F, F)
  do { Count[L1, L2] ← |Count[L1, L2] − T/2|
       if Count[L1, L2] > max
         then { max ← Count[L1, L2]
                maxkey ← (L1, L2)
output (maxkey)
```

Рис. 6.

Задание по проведению линейного криптоанализа шифра, основанного на структуре SPN

Требования

Провести линейный криптоанализ шифра с вашими вариантами таблиц замен и перестановок с целью определения от 8 и более бит пятого раундового подключа. Варианты указаны в приложении. Оформить полученные результаты в виде отчета.

В отчете:

1. привести таблицу замен и таблицу перестановок для вашего варианта;
2. привести таблицу линейных приближений блока замены для своего варианта (как в табл.4 документа «Линейный криптоанализ»);
3. показать, как рассчитываются значения в этой таблице на примере случайной величины с максимальным отклонением (как на рис. 4 документа «Линейный криптоанализ»);
4. построить линейное приближение шифра для своего варианта (вывод формульных соотношений). Показать графически как на рис. 5 документа «Линейный криптоанализ»;
5. привести фрагменты кода программы, которые были изменены для выполнения криптоанализа по вашему варианту;
6. вставить скриншоты результатов работы программы.
7. сделать вывод по итогам проведения линейного криптоанализа

Приложение. Варианты

(Sn, S – означает таблицу замен, n – номер варианта; Pn, P – означает таблицу перестановок, n – номер варианта; номер варианта соответствует номеру студента в списке группы)

S1=[3, 2, 6, 4, 7, 10, 8, 5, 11, 12, 13, 9, 15, 0, 1, 14]

P1=[8, 15, 10, 4, 7, 12, 13, 5, 11, 3, 0, 9, 14, 2, 6, 1]

S2=[3, 8, 13, 4, 14, 9, 10, 12, 5, 6, 0, 1, 11, 2, 7, 15]

P2=[2, 5, 6, 8, 4, 14, 0, 7, 11, 10, 12, 1, 15, 9, 3, 13]

S3=[12, 1, 9, 0, 10, 8, 3, 7, 13, 6, 11, 5, 15, 14, 2, 4]

P3=[5, 11, 1, 13, 2, 15, 0, 8, 3, 6, 12, 7, 9, 14, 4, 10]

S4=[0, 15, 9, 13, 11, 5, 7, 2, 12, 3, 8, 1, 6, 4, 14, 10]

P4=[12, 3, 1, 9, 15, 6, 0, 5, 10, 11, 8, 7, 2, 14, 4, 13]

S5=[6, 8, 13, 1, 5, 10, 2, 11, 15, 12, 9, 0, 14, 3, 7, 4]

P5=[4, 6, 3, 11, 7, 10, 15, 9, 14, 1, 2, 0, 8, 5, 12, 13]

S6=[6, 13, 2, 9, 15, 0, 8, 12, 14, 10, 4, 7, 11, 1, 3, 5]

P6=[4, 7, 9, 15, 12, 8, 3, 6, 11, 0, 5, 14, 1, 2, 10, 13]

S7=[0, 4, 1, 13, 6, 9, 5, 11, 12, 2, 15, 8, 14, 10, 3, 7]

P7=[15, 9, 0, 13, 11, 8, 1, 14, 4, 7, 3, 2, 10, 5, 6, 12]

S8=[4, 5, 7, 8, 2, 3, 12, 9, 15, 11, 1, 0, 14, 10, 13, 6]

P8=[4, 14, 0, 8, 10, 13, 5, 15, 6, 11, 2, 7, 9, 1, 12, 3]

S9=[10, 1, 0, 11, 6, 8, 5, 13, 3, 14, 2, 15, 7, 12, 9, 4]

P9=[3, 11, 10, 0, 5, 1, 13, 4, 8, 14, 2, 12, 6, 9, 7, 15]

S10=[6, 0, 3, 15, 10, 12, 13, 14, 7, 11, 5, 4, 9, 1, 8, 2]
P10=[14, 13, 0, 11, 2, 10, 4, 7, 12, 3, 1, 15, 8, 5, 6, 9]

S11=[2, 15, 0, 4, 5, 9, 8, 11, 6, 3, 14, 1, 13, 12, 10, 7]
P11=[9, 7, 2, 13, 15, 14, 11, 8, 3, 10, 0, 1, 4, 12, 6, 5]

S12=[10, 8, 13, 5, 1, 15, 3, 12, 7, 9, 11, 0, 4, 14, 6, 2]
P12=[9, 11, 4, 14, 0, 7, 5, 13, 15, 3, 1, 8, 12, 6, 10, 2]

S13=[15, 7, 10, 2, 13, 12, 0, 6, 3, 14, 9, 1, 11, 4, 5, 8]
P13=[7, 6, 3, 4, 5, 15, 10, 2, 11, 9, 0, 13, 12, 8, 14, 1]

S14=[2, 0, 7, 4, 6, 1, 12, 5, 13, 3, 14, 15, 8, 9, 11, 10]
P14=[2, 14, 0, 10, 3, 15, 13, 8, 12, 1, 7, 6, 5, 11, 4, 9]

S15=[8, 3, 5, 2, 15, 10, 4, 11, 0, 13, 12, 7, 9, 14, 1, 6]
P15=[8, 11, 5, 12, 9, 13, 1, 14, 6, 15, 4, 10, 3, 7, 2, 0]

S16=[8, 13, 0, 1, 5, 9, 10, 12, 15, 3, 2, 7, 11, 6, 14, 4]
P16=[4, 14, 15, 1, 11, 7, 12, 6, 13, 3, 9, 2, 0, 8, 5, 10]

S17=[6, 14, 1, 7, 11, 0, 4, 13, 8, 15, 9, 5, 2, 12, 10, 3]
P17=[6, 2, 14, 0, 8, 10, 11, 4, 9, 5, 3, 15, 7, 12, 1, 13]

S18=[12, 11, 7, 15, 6, 2, 1, 13, 14, 8, 0, 10, 4, 5, 3, 9]
P18=[7, 4, 2, 6, 15, 5, 0, 10, 8, 11, 3, 14, 12, 13, 9, 1]

S19=[6, 2, 12, 3, 1, 7, 0, 15, 4, 10, 14, 9, 5, 8, 13, 11]

P19=[6, 11, 13, 3, 9, 10, 14, 7, 15, 2, 1, 8, 4, 12, 0, 5]

S20=[0, 5, 9, 8, 4, 6, 14, 2, 1, 3, 7, 11, 13, 10, 12, 15]

P20=[5, 0, 3, 7, 15, 12, 2, 6, 13, 9, 11, 1, 10, 8, 14, 4]

S21=[6, 3, 15, 10, 0, 4, 2, 12, 9, 5, 13, 8, 11, 7, 14, 1]

P21=[15, 14, 3, 7, 6, 11, 8, 0, 12, 10, 9, 5, 13, 4, 2, 1]

S22=[14, 3, 6, 11, 0, 1, 12, 15, 5, 9, 8, 7, 13, 4, 10, 2]

P22=[0, 9, 4, 15, 8, 5, 14, 12, 3, 11, 2, 1, 7, 13, 10, 6]

S23=[5, 13, 4, 15, 11, 2, 9, 8, 10, 12, 14, 7, 0, 6, 1, 3]

P23=[13, 2, 1, 10, 3, 5, 0, 14, 9, 7, 11, 4, 6, 8, 15, 12]

S24=[9, 12, 15, 1, 0, 2, 10, 8, 14, 7, 6, 3, 11, 13, 4, 5]

P24=[15, 8, 0, 13, 6, 5, 14, 9, 2, 11, 10, 3, 7, 12, 4, 1]

Литература

[1] M. Matsui. Linear cryptanalysis method for DES cipher. In Advances in Cryptology - EUROCRYPT’93, volume 765 of LNCS, pages 386–397. Springer-Verlag, 1993.

[2] M. Matsui. The first experimental cryptanalysis of the Data Encryption Standard. In Advances in Cryptology - CRYPTO’94, volume 839 of LNCS, pages 1–11. Springer-Verlag, 1994.

[3] Douglas R. Stinson. Cryptography: Theory and Practice, Third Edition (Discrete Mathematics and Its Applications), p.616, 2005.