TwoFish

Криптоалгоритм $Twofish$[1] шифрует 128-битовые блоки открытых данных под управлением секретного ключа, длина которого может составлять $N = 128, 192$ или 256 битов (далее рассматривается случай $N = 128$; более короткие ключи дополняются до требуемой длины нулями.) По своей структуре $Twofish$ является классическим шифром Фейстеля. Рекомендуемое число раундов шифрования $R = 16$.

Блок P шифруемых данных, являющийся массивом из шестнадцати байтов: $P = (p_0, p_1, \dots, p_{15})$, представляется так же в виде четырёх 4-байтовых слов P_0, P_1, P_2 и P_3 с прямым порядком байтов ($little\text{-}endian$), т.е.

$$P_i = p_{4i} + p_{4i+1}2^8 + p_{4i+2}2^{16} + p_{4i+3}2^{24},\ i = 0, 1, 2, 3.$$

Далее заглавные буквы обозначают 4-байтовые слова, а строчные – составляющие их байты. Например, $X = (x_0, x_1, x_2, x_3)$ – слово с байтами x_0, x_1, x_2, x_3, начиная с младшего.

В алгоритме используются следующие операции и преобразования:

$X \oplus Y$ – побитовое сложение X и Y по модулю 2;
$X + Y$ – сложение X и Y по модулю 2^{32}.
$rol_s(X)$ $(ror_s(X))$ – циклический сдвиг слова X на s битов влево (вправо).

Псевдоадамарово преобразование PHT определяется как

$$PHT(X, Y) \equiv \{X := X + Y;\ Y := X + Y\}.$$

Функция $MDS(X)$ возвращает значение Y, определяемое как

$$\begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 0\text{x}01 & 0\text{x}ef & 0\text{x}5b & 0\text{x}5b \\ 0\text{x}5b & 0\text{x}ef & 0\text{x}ef & 0\text{x}01 \\ 0\text{x}ef & 0\text{x}5b & 0\text{x}01 & 0\text{x}ef \\ 0\text{x}ef & 0\text{x}01 & 0\text{x}ef & 0\text{x}5b \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix},$$

причем байты интерпретируются как элементы конечного поля $\mathbb{F}_{256} \cong \mathbb{F}_2[x]/v(x)$, где $v(x) = x^8 + x^6 + x^5 + x^3 + 1$ – примитивный многочлен 8-ой степени над полем $\mathbb{F}_2$.

Функция $R(X, Y)$ возвращает значения Z, определяемое как

$$\begin{pmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \end{pmatrix} = \begin{pmatrix} 0\text{x}01 & 0\text{x}a4 & 0\text{x}55 & 0\text{x}87 & 0\text{x}5a & 0\text{x}58 & 0\text{x}db & 0\text{x}9e \\ 0\text{x}a4 & 0\text{x}56 & 0\text{x}82 & 0\text{x}f3 & 0\text{x}1e & 0\text{x}c6 & 0\text{x}68 & 0\text{x}e5 \\ 0\text{x}02 & 0\text{x}a1 & 0\text{x}fc & 0\text{x}c1 & 0\text{x}47 & 0\text{x}ae & 0\text{x}3d & 0\text{x}19 \\ 0\text{x}a4 & 0\text{x}55 & 0\text{x}87 & 0\text{x}5a & 0\text{x}58 & 0\text{x}db & 0\text{x}9e & 0\text{x}e3 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix},$$

причем в данном случае байты интерпретируются, как элементы конечного поля $\mathbb{F}_{256} \cong \mathbb{F}_2[x]/w(x)$, где $w(x) = x^8 + x^6 + x^3 + x^2 + 1$ – примитивный многочлен 8-ой степени над полем $\mathbb{F}_2$.

Подстановки (перестановки) q_0 и q_1 заданы на множестве байтов. Для байта x значение $y = q_i[x]$, $i = 0{,}1$, вычисляется по схеме:

$(a, b) := (x\ div\ 16, x\ mod\ 16)$;
$(c, d) := (a \oplus b, a \oplus (b\ div\ 2) \oplus ((8 * b) mod\ 16) \oplus ((8 * a) mod\ 16))$;
$(a, b) := (t_0^{(i)}[c], t_1^{(i)}[d])$;
$(c, d) := (a \oplus b, a \oplus (b\ div\ 2) \oplus ((8 * b) mod\ 16) \oplus ((8 * a) mod\ 16))$;
$(a, b) := (t_2^{(i)}[c], t_3^{(i)}[d])$;
$y := 16 * b + a$.

[1] Авторы шифра: *Bruce Schneider*, *John Kelsey*, *Doug Whiting*, *Chris Hall* и *Niels Ferguson* (США)

Здесь a, b, c, d – вспомогательные переменные (байты), а значения $t_j^{(i)}[u]$ заданы следующей таблицей (в 16-ичном представлении):

u	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$t_0^{(0)}$	8	1	7	d	6	f	3	2	0	b	5	9	e	c	a	4
$t_1^{(0)}$	e	c	b	8	1	2	3	5	f	4	a	6	7	0	9	d
$t_2^{(0)}$	b	a	5	e	6	d	9	0	c	8	f	3	2	4	7	1
$t_3^{(0)}$	d	7	f	4	1	2	6	e	9	b	3	0	8	5	c	a
$t_0^{(1)}$	2	8	b	d	f	7	6	e	3	1	9	4	0	a	c	5
$t_1^{(1)}$	1	e	2	b	4	c	3	7	6	d	a	5	f	9	0	8
$t_2^{(1)}$	4	c	7	5	1	6	9	a	0	e	d	8	2	b	3	f
$t_3^{(1)}$	b	9	5	1	c	3	d	e	6	4	7	f	2	0	8	a

Функция $h(X, Y, Z)$ возвращает значение $W = MDS(U)$,где $U = (u_0, u_1, u_2, u_3)$ определяется как:

$u_0 := q_0[z_0 \oplus q_1[y_0 \oplus q_1[x_0]]]$;
$u_1 := q_1[z_1 \oplus q_1[y_1 \oplus q_0[x_1]]]$;
$u_2 := q_0[z_2 \oplus q_0[y_2 \oplus q_1[x_2]]]$;
$u_3 := q_1[z_3 \oplus q_0[y_3 \oplus q_0[x_3]]]$.

Раундовая функция $F[RK_0, RK_1, S_0, S_1](R_0, R_1)$ с 4-байтовыми ключевыми параметрами RK_0, RK_1, S_0 и S_1 и аргументами R_0 и R_1 возвращает 4-байтовые значения F_0 и F_1, определяемые как:

$(F_0, F_1) := (h(R_0, S_0, S_1), h(rol_8(R_1), S_0, S_1))$;
PHT(F_0, F_1);
$(F_0, F_1) := (F_0 + RK_0, F_1 + RK_1)$.

Вычисление раундовых подключей

На основе 16-байтового секретного ключа $K = (k_0, k_1, \dots, k_{15})$ вычисляются 4-байтовые подключи S_0 и S_1, используемые в определении функции F и раундовые подключи K_0, K_1,…, K_{39} (для R-раундового алгоритма необходимо $2R + 8$ подключей (здесь в качестве стандартного значения принято $R = 16$):

for $i := 0$ **to** 3 **do** $M_i = (k_{4i}, k_{4i+1}, k_{4i+2}, k_{4i+3})$;
$S_0 := RS(M_0, M_1)$;
$S_1 := RS(M_2, M_3)$;
$(X, Y, Z) := (0, 2^{24} + 2^{16} + 2^8 + 1, 2^{25} + 2^{17} + 2^9 + 2)$;
for $i := 0$ **to** 19 **do** {
    $A := h(X, M_2, M_0)$;
    $B := rol_8(h(Y, M_3, M_1))$;
    $PHT(A, B)$;
    $(K_{2i}, K_{2i+1}) := (A, rol_9(B))$;
    $X := X + Z$;
    $Y := Y + Z$
}.

Алгоритм зашифрования *Twofish*

Вход: $P = (P_0, P_1, P_2, P_3)$ – 128-блок открытых данных, представленный в виде четырех 4-байтовых слов P_0, P_1, P_2 и P_3.

1. (Входное забеливание.)

$$(C_0, C_1, C_2, C_3) := (P_0 \oplus K_0, P_1 \oplus K_1, P_2 \oplus K_2, P_3 \oplus K_3);$$

2. (16-раундов, или 8 циклов зашифрования).

$\boldsymbol{for}\ i := 0\ \boldsymbol{to}\ 7\ \boldsymbol{do}\ \{$

$(F_0, F_1) := F[K_{4i+8}, K_{4i+9}, S_0, S_1](C_0, C_1);$

$(C_2, C_3) := (rol_1(C_2 \oplus F_0), rol_1(C_3) \oplus F_1);$

$(F_0, F_1) := F[K_{4i+10}, K_{4i+11}, S_0, S_1](C_2, C_3);$

$(C_0, C_1) := (rol_1(C_0 \oplus F_0), rol_1(C_1) \oplus F_1)$

$\};$

$C_0 \leftrightarrow C_2;$

$C_1 \leftrightarrow C_3;$

3. (Выходное забеливание.)

$(C_0, C_1, C_2, C_3) := (C_0 \oplus K_4, C_1 \oplus K_5, C_2 \oplus K_6, C_3 \oplus K_7)$.

Выход: $C = (C_0, C_1, C_2, C_3)$ – 128-битовый блок шифртекста.

Алгоритм расшифрования *Twofish*

Вход: $C = (C_0, C_1, C_2, C_3)$ – 128-битовый блок шифртекста, представленный в виде четырех 4-байтовых слов C_0, C_1, C_2 и C_3.

$P := (C_0 \oplus K_4, C_1 \oplus K_5, C_2 \oplus K_6, C_3 \oplus K_7);$

$\boldsymbol{for}\ i := 7\ \boldsymbol{downto}\ 0\ \boldsymbol{do}\ \{$

$(F_0, F_1) := F[K_{4i+10}, K_{4i+11}, S_0, S_1](P_0, P_1);$

$(P_2, P_3) := (rol_1(P_2) \oplus F_0, rol_1(P_3 \oplus F_1);$

$(F_0, F_1) := F[K_{4i+8}, K_{4i+9}, S_0, S_1](P_2, P_3);$

$(P_0, P_1) \coloneqq (rol_1(P_0) \oplus F_0, rol_1(P_1 \oplus F_1)$

$\};$

$P_0 \leftrightarrow P_2;$

$P_1 \leftrightarrow P_3;$

$(P_0, P_1, P_2, P_3) := (P_0 \oplus K_0, P_1 \oplus K_1, P_2 \oplus K_2, P_3 \oplus K_3)$.

Выход: $P = (P_0, P_1, P_2, P_3)$ – 128-битовый блок открытых данных.

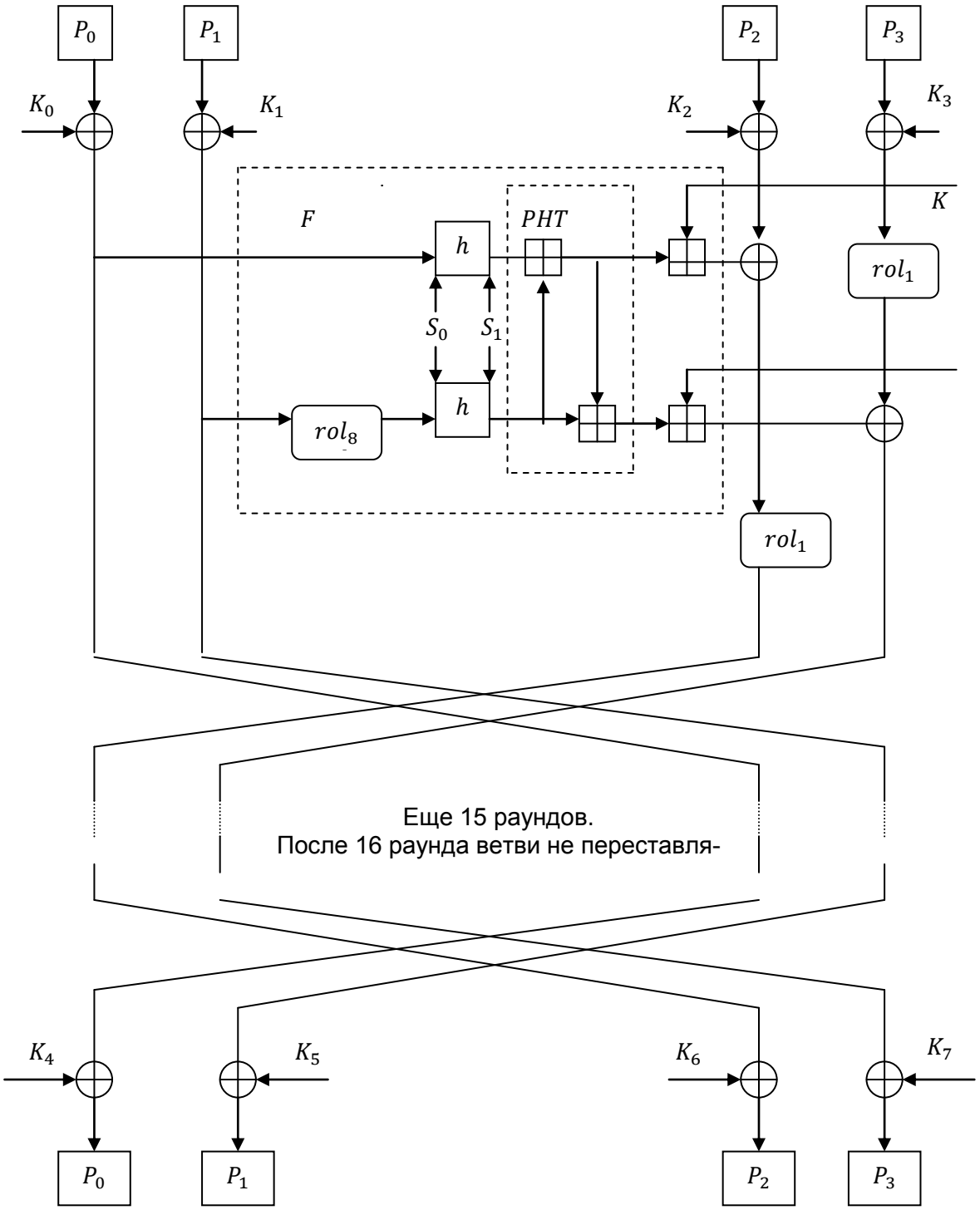

Рис.1. *Структура алгоритма* $Twofish$