$DFCv2$

Криптоалгоритм $DFCv2$ ($Decorrelated\ Fast\ Cipher\ version\ 2$) [1] характеризуется следующими параметрами:

m – длина блока шифруемых данных (в битах),

k – длина секретного ключа (в битах),

r – число раундов шифрования,

s – число раундов при генерации раундовых подключей.

Эти параметры должны удовлетворять ограничениям: $m \geq 32$, $0 \leq k \leq 2m$, $rs \leq 128$, m кратно 4, r четно. Соответствующий шифр обозначается $DFCv2\ (m, k, r, s)$. Здесь рассматривается шифр со стандартными значениями параметров: $m = 128$, $k \in \{128, 192, 256\}$, $r = 8$ и $s = 4$.

По своей структуре $DFCv2$ – классический шифр Фейстеля. Далее, в описании алгоритма, $RF(X, RK)$ обозначает раундовую функцию шифрования; ее аргументами являются: X – 64-битовая левая половина шифруемого блока – и RK – 128-битовый раундовый подключ. Функция RF возвращает 64-битовое значение. Используемые раундовые подключи RK_1, …, RK_8 генерируются на основе секретного ключа K также по схеме Фейстеля.

Алгоритм зашифрования

Вход: PT – 128-битовый блок открытых данных.

$X \coloneqq PT.L;$

$Y := PT.R;$

$\boldsymbol{for}\ i \coloneqq 1\ \boldsymbol{to}\ 8\ \boldsymbol{do}\ \{$

$\quad Y \coloneqq Y \oplus RF(X, RK_i);$

$\quad X \leftrightarrow Y$

$\};$

Выход: $CT = Y \parallel X$ – 128-битовый блок шифртекста.

Этот же алгоритм используется и для расшифрования, но с обратным порядком использования раундовых подключей.

Определение раундовой функции

Функция RF ($Round\ Function$) определяется как

$$RF(x, k) = CP(((k.L \times x + k.R)\ mod\ p) \bmod 2^{64}).$$

Здесь $p = 2^{64} + 13$ – наименьшее простое число, превосходящее 2^{64}. Функция CP ($Confusion\ \ Permutation$) задана как

$$CP\colon \mathrm{B}_2^{64} \to \mathrm{B}_2^{64},\ CP(y) = ((y.R \oplus RT_s) \parallel (y.L \oplus KC)) \boxplus_{64} KD,$$

где $s = trunc_6(y.L)$ – индекс, числовое значение которого определяется 6 левыми (старшими) битами 32-битового блока $y.\mathrm{L}$. Используемые в определении функции CP 32-битовые константы RT_0, …, RT_{63}, KC и 64-битовая константа KD определены ниже.

Вычисление раундовых подключей

Раундовые подключи RK_1, …, RK_8 вычисляются последовательно путем 8-кратного применения итерационной схемы Фейстеля. При этом используются вспомогательные переменные:

PD – 256-битовая переменная ($Padded\ Key$),

$IRK = (\mathrm{IRK}_1, \ldots, \mathrm{IRK}_{32})$ – массив 128-битовых переменных ($Internal\ Round\ Key$),

IRK_0, RK_0 – 128-битовые переменные (начальное значение при генерации массивов IRK и RK)

и константы (кроме упомянутых ранее RT_0, …, RT_{63}):

KS – 256-битовая константа,

---

[1] Авторы шифра: *Louis Granboulan*, *Phong Q. Nguyen*, *Fabrice Noilhan* (Франция), *Serge Vaudenay* (Швейцария)

$KAB = (KAB_0, \ldots, KAB_{15})$ – массив 128 битовых констант.

1. (Ключ K дополняется до длины 256 левыми битами константы KS.)
   $PK := K \parallel trunc_{256-k} KS$;
2. (Инициализируются переменные IRK_0 и RK_0.)
   $IRK_0 := PK.L$;
   $\mathrm{RK}_0 :=$ PK. R;

(Вычисляются внутренние раундовые подключи.)

for $i := 1$ ***to*** 15 ***do*** {
   $s := RT_i \bmod 16$;
   $IRK_i := IRK_{i-1} \oplus KAB_s$
};

3. (8-кратный цикл для вычисления раундовых подключей $RK_1, \ldots, RK_8$.)

for $i := 1$ ***to*** 8 ***do*** {

3.1. Вычисление согласно 4-раундовой схеме Фейстеля очередного раундового подключа RK_i, исходя из подключа RK_{i-1}:

$X := RK_{i-1}.L$;
$Y := RK_{i-1}.R$;
for $j := 1$ ***to*** 4 ***do*** {
   $Y := Y \oplus RF(X, IRK_{4(i-1)+j})$;
   $X \leftrightarrow Y$};
   $RK_i := Y \parallel X$
}.

Определение констант, используемых в $DFCv2(128, k, 8,4)$

Пусть $E = (E_0, E_1, \ldots, E_{71})$ − массив 32-битовых слов, образованный 16-ичными цифрами дробной части числа e – основания натуральных логарифмов (см. табл.1):

$$e = \sum_{n=0}^{\infty} \frac{1}{n!} = (2.71 \ldots)_{10} = (2.\mathrm{B7E15162} \ldots \mathrm{144E49C2})_{16}$$

Необходимые константы определяются следующим образом:

$RT_{[0..63]} = E_{[0..63]}$; $KD = E_{64} \parallel E_{65}$; $KC = E_{66}$;
$KAB_i = E_{4i} \parallel E_{4i+1} \parallel E_{4i+2} \parallel E_{4i+3}$, $i = 0, 1, \ldots, 15$;
$KS = E_{64} \parallel E_{65} \parallel E_{66} \parallel E_{67} \parallel E_{68} \parallel E_{69} \parallel E_{70} \parallel E_{71}$.
for $i := 0$ ***to*** 63 ***do*** $RT_i := E_i$;
$KD := E_{64} \parallel E_{65}$;
$KC := E_{66}$;
for $i := 0$ ***to*** 15 ***do*** $KAB_i := E_{4i} \parallel E_{4i+1} \parallel E_{4i+2} \parallel E_{4i+3}$;
$KS :=$ $E_{64} \parallel E_{65} \parallel E_{66} \parallel E_{67} \parallel E_{68} \parallel E_{69} \parallel E_{70} \parallel E_{71}$.

Таблица 1

b7e15162	8aed2a6a	bf715880	9cf4f3c7	62e7160f	38b4da56
a784d904	5190cfef	324e7738	926cfbe5	f4bf8d8d	8c31d763
da06c80a	bb1185eb	4f7c7b57	57f59584	90cfd47d	7c19bb42
158d9554	f7b46bce	d55c4d79	fd5f24d6	613c31c3	839a2ddf
8a9a276b	cfbfa1c8	77c56284	dab79cd4	c2b3293d	20e9e5ea
f02ac60a	cc93ed87	4422a52e	cb238fee	e5ab6add	835fd1a0
753d0a8f	78e537d2	b95bb79d	8dcaec64	2c1e9f23	b829b5c2
780bf387	37df8bb3	00d01334	a0d0bd86	45cbfa73	a6160ffe
393c48cb	bbca060f	0ff8ec6d	31beb5cc	eed7f2f0	bb088017
163bc60d	f45a0ecb	1bcd289b	06cbbfea	21ad08e1	847f3f73
78d56ced	94640d6e	f0d3d37b	e67008e1	86d1bf27	5b9b241d
eb64749a	47dfdfb9	6632c3eb	061b6472	bbf84c26	144e49c2