Nimbus

Криптоалгоритм $Nimbus$ [1] шифрует 64-битовые блоки открытых данных под управлением секретного ключа, длина которого может составлять от 64 до 576 битов (с шагом 64).

Алгоритм состоит из пяти раундов, в каждом из которых используются только три примитива: умножение по модулю 2^{64} ($\boxtimes_{64}$), побитовое сложение по модулю 2 ($\oplus$) и операция замены порядка битов в блоке на обратный (блок, получающийся из блока X путем перестановки -го и $(63-n)$-го битов, $n = 0, 1, \ldots, 31$, обозначается как $g(x)$). Функция зашифрования 64-битового блока X под управлением 64-битовых раундовых подключей k_0, k_1, …, k_9 определяется как

$Nimbus\,[k_0, k_1, \ldots, k_9](X) \equiv \{$
for $i := 0$ **to** 4 **do** $x := k_i \boxtimes_{64} g(k_{i+5} \oplus X)$;
$return\ (X)$
}.

Раундовые подключи k_0, k_1, k_2, k_3 и k_4 должны быть нечетными числами; в этом случае обратное преобразование существует и имеет вид:

$Nimbus^{-1}\,[k_0, k_1, \ldots, k_9](X) \equiv \{$
for $i := 4$ **downto** 1 **do** $x := g(k^{-1} \boxtimes_{64} X) \oplus k_{i+4}$;
$return\ (X)$
}.

Здесь k^{-1} – мультипликативный обратный по модулю 2^{64} к k, т.е. $k^{-1} \boxtimes_{64} k = 1$.

Вычисление раундовых подключей

Подключи k_0, k_1, …, k_n генерируются на основе последовательности чисел s_0, s_1, …, s_9, где n нечетно и $m < n$. (Последовательность s_i является секретным ключом. В рассмотренном варианте шифра $n = 9$, но здесь приводится более общая схема, когда число раундов равно $n\ div\ 2$.) При вычислении k_i используются 64-битовые константы c_0, c_1, …, c_9, образованные дробной частью числа π; в 16-ичном представлении они имеют следующие значения:

$243F6A8885A308D3$	$13198A2E03707345$	$A4093822299F31D1$
$082EFA98EC4E6C89$	$452821E638D01377$	$BE5466CF34E90C6C$
$C0AC29B7C97C50DD$	$3F84D5B5B5470917$	$9216D5D98979FB1B$
$D1310BA698DFB5AC$		

Отметим, что c_0, c_1, c_2, c_3 и c_4 – нечетные числа.

Псевдокод генерации k_j из s_i имеет вид:

for $j := 0$ **to** n **do** $k_j := 0$;
for $i := 0$ **to** m **do** {
$y := s_i \oplus Nimbus[c_0, c_1, \ldots, c_9](s_i)$;
for $j := 0$ **to** n **do** {
$y := Nimbus[c_0, c_1, \ldots, c_9](y)$;
$k_i = y \oplus Nimbus[c_0, c_1, \ldots, c_9](y \oplus k_i)$
}
};
for $j := 0$ **to** $(n$ **div** $2)$ **do** $k_j := k_j \vee 1$.

Вычисление мультипликативного обратного по модулю 2^m

Замечание: Алгоритм $Nimbus$ нетрудно обобщить на случай, когда шифруются блоки другой длины 2^m ($m = 4, 5, 7, 8, \ldots$).

Значение $y = a^{-1} mod\ 2^m = a^{2^{m-1}-1} mod\ 2^m$ можно вычислить по схеме:
$y := 1$; **for** $i := 0$ **to** $m - 1$ **do** $\{y := y \cdot a\ mod\ 2^m;\ a := (a \cdot a)\ mod\ 2^m\}$

---

[1] Автор шифра: *Alexis Warner Machado* (Бразилия)