1. ARC4

1.1. Теоретические сведения

ARC4 – потоковый шифр с переменным размером ключа. Алгоритм работает в режиме обратной связи по выходу: поток ключей не зависит от открытого текста.

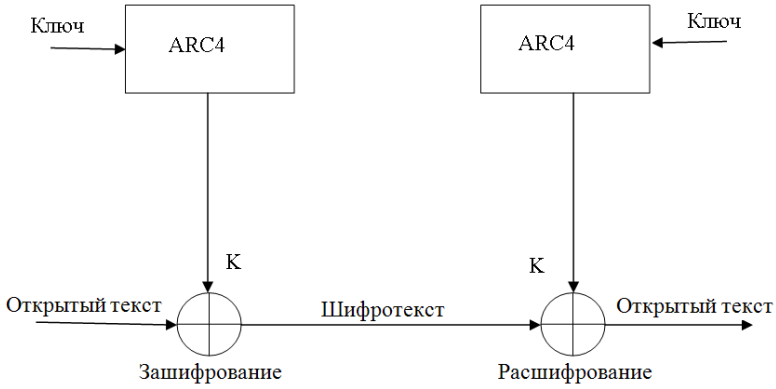

Рисунок 1. Схема поточного шифра

В классической реализации используется S-блок размером 8*8: $S = \{S_0, S_1, ..., S_{255}\}$. Элементы представляют собой перестановку чисел от 0 до 255, а перестановка является функцией ключа переменной длины. В алгоритме используются 2 счетчика: $i = 0,\ j = 0$.

1.1.1. Инициализация S-блока

Алгоритм также известен как «key-scheduling algorithm» или «KSA». Заполняется массив $S = \{0,1,2,...255\}$. Также заполняется ключем другой массив, при необходимости для заполнения всего массива повторяя ключ $Key = \{\mathrm{Key}_0, \mathrm{Key}_1, ..., \mathrm{Key}_{255}\}$. Устанавливаем значение индекса $j = 0$. Затем:

$for\ i = 0..255:$

$j = (j + S_i + Key_i) \bmod 256$

$swap(S_i, S_j)$ //*Меняем местами* S_i, S_j

1.1.2. Генерация псевдослучайного слова K

Эта часть алгоритма называется генератором псевдослучайной последовательности (англ. pseudo-random generation algorithm, PRGA).

$i = 0,\ j = 0$

Для генерации случайного байта выполняется следующее:

$$i = (i+1) \bmod 256;\ \ j = (j + S_i) \bmod 256$$

$$swap(S_i, S_j)$$

$$t = (S_i + S_j) \bmod 256$$

$$K = S_t$$

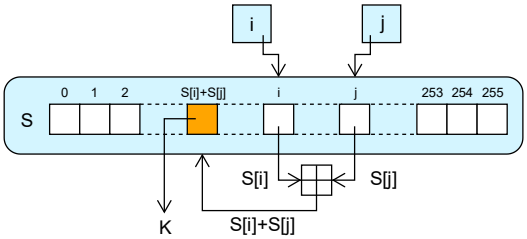

Рисунок 2. Генерация K в графическом виде

Байт K используется в операции Xor с открытым текстом, для получения шифротекста или в операции Xor с шифротекстом для получения открытого текста.

1.1.3. Стойкость

RC4 может находиться $256! \cdot 256^2 \approx 2^{1700}$ возможных состояний. S-блок медленно изменяется при использовании: i обеспечивает изменение каждого элемента, а j -что элементы изменяются случайным образом.

1.1.4. Преимущества

Шифрование выполняется примерно в 10 раз быстрее, чем DES.