УДК 519.6 **Технические науки**

Дедов Олег Петрович, *кандидат технических наук, доцент,*
МИРЭА-Российский технологический университет
кафедра КБ-1 «Защита информации»

ПРИМЕНЕНИЕ МАТРИЧНОГО ПОДХОДА К ПОТОЧНОМУ ШИФРОВАНИЮ

Аннотация: цель данной работы состоит в вычислении последовательности псевдослучайных чисел, то есть гаммы с помощью матричного алгоритма и применение данной последовательности в поточном шифровании.

Ключевые слова: гамма, квадратная матрица, матрица – строка, матрица-столбец, модуль.

Abstract: The purpose of this paper is to calculate a sequence of pseudorandom numbers, i.e., gamma, using a matrix algorithm and apply this sequence in stream encryption.

Keywords: gamma, square matrix, row matrix, column matrix, module.

Введение

В настоящее время в системах связи широкое применение нашли алгоритмы поточного шифрования, которые в отличие от блочного шифрования выполняют преобразование информации в текущий момент времени i, поступающей от источника сообщений x_i , по одному биту путём сложения x_i по модулю два со значением k_i , вырабатываемым генератором ключей, и также имеющим значение один бит. Таким образом, поточный шифр устраняет необходимость разбивать исходное сообщение (открытый текст) на блоки одинаковой длины по 64 бит или по 128 бит, как это

делается в алгоритмах блочного шифрования, таких как DES или AES. Следовательно, поточный шифр может работать в реальном времени и при передаче сообщения, представленного в виде последовательности цифр в двоичном коде, то есть в виде нулей и единиц, каждая цифра может шифроваться и передаваться мгновенно. Схема работы поточного шифра представлена на рис. 1.1.

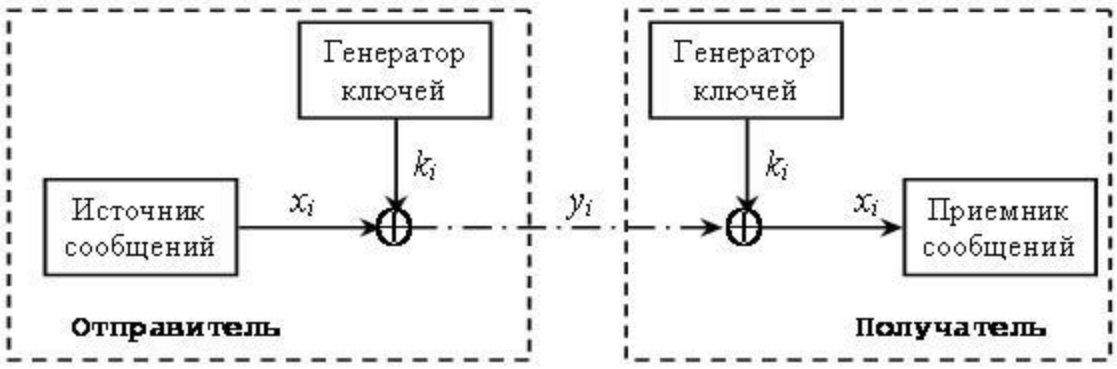

Рис. 1.1. Схема работы поточного шифра

Рассмотрим работу поточного шифра, представленного на рис. 1.1.На стороне, как отправителя информации, так и получателя информации находится генератор ключей, который генерирует одинаковую последовательность битов $k_1, k_2,..., k_n$, называемую гаммой. Источник сообщений генерирует биты открытого текста x_i, которые складываются по модулю два с гаммой, в результате чего получаются биты зашифрованного сообщения y_i:

$$y_i = x_i \oplus k_i, i = 1, 2, \ldots, n$$

Чтобы из зашифрованного текста $y_1, y_2,..., y_n$ восстановить сообщение $x_1, x_2,..., x_n$, надо использовать для расшифрования формулу, то есть к значениям $y_1, y_2,..., y_n$ прибавить последовательность битов $k_1, k_2,..., k_n$

$$x_i = y_i \oplus k_i, \ i = 1, 2, \ldots, n$$

Рассмотрим пример, который покажет шифрование и расшифрование информации с применением гаммы. Предположим, нам необходимо зашифровать десятичное число 15 методом гаммирования с использованием

ключа 10. Для этого вначале необходимо преобразовать исходное число и ключ (гамму) из десятичной формы записи в двоичную форму, то есть $15_{10}=1111_2$, $10_{10}=1010_2$. Затем надо записать полученные двоичные числа друг под другом и каждую пару символов сложить по модулю два. Известно, что при сложении двух двоичных цифр (ноль или единица) по модулю два - получается ноль, если исходные двоичные цифры одинаковы, то есть равны либо нулю, либо единице и единице. Если же цифры разные, то есть ноль плюс единица или единица плюс ноль, то в результате получим единицу.

Например: Пусть надо найти сумму двух десятичных чисел 15 (исходный текст) и 10(гамма). Запишем эти числа в двоичной системе счисления и сложим по модулю два двоичных числа 1111(исходное число -15) и 1010(гамма-10):

Исходное число 1 1 1 1

Гамма 1 0 1 0

Результат 0 1 0 1

В результате сложения получается двоичное число 0101. Если перевести его в десятичную форму, получим цифру 5(пять). Таким образом, в результате сложение числа 15 с гаммой 10 получаем число 5. Для расшифрования необходимо зашифрованное число 5 представляется в двоичном виде как 0101 и произвести сложение по модулю 2 с ключом, то есть с числом 10 в двоичном виде 1010:

Зашифрованное число 0 1 0 1

Гамма 1 0 1 0

Результат 1 1 1 1

Переведем полученное двоичное значение 1111 в десятичный вид и получим число 15, то есть исходное число. Таким образом, при гаммировании по модулю 2 можно использовать одну и ту же операцию, как для шифрования, так и для расшифрования. Это позволяет использовать один и тот же алгоритм, а соответственно и одну и ту же программу при программной реализации, как для шифрования, так и для расшифрования.

Операция сложения по модулю два очень быстро выполняется на

компьютере (в отличие от многих других арифметических операций), поэтому наложение гаммы даже на очень большой открытый текст выполняется практически мгновенно. Данный подход к шифрованию и расшифрованию получл название метода гаммирования.

Благодаря указанным достоинствам метод гаммирования широко применяется в современных технических системах сам по себе, а также как элемент комбинированных алгоритмов шифрования.

Сформулируем, как производится гаммирование по модулю 2 в общем случае:

1. Символы исходного текста и гамма представляются в двоичном коде и располагаются один под другим, при этом ключ (гамма) записывается столько раз, сколько потребуется;

2.Каждая пара двоичных знаков складывается по модулю два;

3.Полученная последовательность двоичных знаков кодируется символами алфавита в соответствии с выбранным кодом.

При использовании метода гаммирования, ключом является последовательность чисел, с которой производится сложение – гамма, то есть последовательность битов $k_1, k_2,..., k_n$. Если гамма короче, чем сообщение, предназначенное для шифрования, то гамма повторяется требуемое число раз.

В настоящее время для выработки гаммы применяются: линейный конгруэнтный генератор псевдослучайных чисел, метод Фибоначчи с запаздыванием, генератор псевдослучайных чисел на основе алгоритма BBS, отдельные режимы работы алгоритма DES, алгоритм RC- или аналогичные шифры, которые вырабатывают последовательность чисел k_i.

Из всех вышеперечисленных наиболее широкое применение нашёл алгоритм поточного шифрования RC4.

Алгоритма RC4 относится к классу алгоритмов, определяемых размером его блока или слова – параметром n [1]. Внутреннее состояние RC4 состоит из последовательности чисел размером 2^n слов, при n=4 количество чисел данной последовательности равено-16 и может быть записано в десятичной форме-

0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15 и двух счетчиков - i-й и j-й счётчики, оба при n=4 являются 4-битовыми. Все вычисления проводятся по модулю 2^n, то есть при n=4-это означает, что вычисления будут проводиться по модулю 16.

В алгоритме RC 4- последовательность чисел используется как *таблица* замен (*S-бокс)*, и обозначается как S и при n=4 массив S будет состоять из 2^4 =16 элементов, как отмечено выше, то есть из чисел 0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15. В каждый момент времени *таблица* S содержит все возможные n-битовые (в нашем случае 4-битовые) числа от 0 до 15 в перемешанном виде. Конкретная перестановка значений в таблице определяется ключом К, который представляет – произвольный набор чисел. Если длина ключа меньше чем 2^n , а в данном случае меньше чем 16, то ключ повторяется столько раз, пока количество цифр не будет равно 2^n. После того как все числа будут перемешаны закончится первый этап, а на втором этапе производится выборка псевдослучайного слова (числа).

Для этого счетчикам i и j присваивается начальное *значение* ноль. Затем для получения каждого нового значения z_i выполняются следующие действия (этап *№2*):

$i = (i + 1) \bmod 16;$

$j = (j + S_i) \bmod 16;$

поменять местами S_i и S_j;

$t = (S_i + S_j) \bmod 16;$

$z_i = S_t$.

Значение z_i вычисляется в десятичной форме записи, то есть как одно из чисел в диапазоне от 0 до 15.Для получения гаммы, то есть k_i, вычисляемые на каждом шаге последовательности чисел z_i записываем в двоичном виде и суммируя побитно по модулю два с открытым текстом x_i получаем зашифрованное значение y_i с применением алгоритм поточного шифрования.

В данной статье предлагается использовать матричный подход к построению как матрицы SQ (аналог матрицы S), так и вспомогательной

матрицы KQ, которая является аналогом ключа в алгоритме RC4 и получать псевдослучайные числа как элементы матрицы SQL каждый элемент которой вычисляется в результате матричных преобразований над матрицами SQ и KQ.

В отличие от алгоритма RC4 , относящегося к классу алгоритмов, определяемых размером его блока или слова – параметром n, в предлагаемом матричном алгоритме определяющим параметром является размер матрицы N, который может быть выражен через n.

Внутреннее состояние в алгоритме RC4 состоит из массива размером 2^n слов, а в предлагаемом матричном алгоритме внутреннее состояние определяется- квадратной матрицы SQ размером N*N, где

$$N^2 = 2^n \quad \text{или } N= 2^{n\backslash 2} , (1)$$

то есть при n=4 N=4, при n=8 N=16, при n=10 N=32 и т.д.

Квадратная матрица SQ, используется как таблица замен S (S-бокс) в RC4. В начальный момент времени матрица SQ также, как и таблица замен S содержит все возможные n-битовые числа, записанные либо по порядку, либо в перемешенном виде, то есть записанные в соответствии с заранее согласованной (между отправителем и получателем) последовательности.

Матричный алгоритм состоит из двух этапов, аналогично алгоритму RC4 . На первом, подготовительном этапе производится *инициализация* матрицы замен SQ и вычисление вспомогательной матрицы KQ, которая является аналогом ключа K в алгоритме RC4, по формуле

$$KQ = L*M \bmod 2^n, (2)$$

где L-матрица порядка N*1, а M порядка 1*N, причём числа входящие в данные матрицы выбираются произвольным образом из чисел, входящих в матрицу SQ, а элементы вспомогательной матрицы KQ вычисляются по модулю 2^n.

На втором, основном этапе вычисляется по модулю 2^n матрица SQL по формуле:

$$SQL = (SQ +KQ) \bmod 2^n \text{ или } SQL =(SQ *KQ) \bmod 2^n \quad (3)$$

и производится выборка псевдослучайных чисел, которыми будут являться элементы данной матрицы SQL, то есть SQLij - это псевдослучайные числа, соответствующие числам ki на рис. 1.1.

Матрицы SQL, SQ и KQ имеют одинаковый порядок, и представляет собой квадратные матрицы размера N*N.

Рассмотрим принцип работы матричного алгоритма при n=4. В этом случае количество всех чисел, входящих в массив S(алгоритм RC4) или в матрицу SQ будет равно $2^4 = 16$, а сами числа будут находиться в диапазоне от 0 до 15. Тогда N, вычисляемое по формуле (1) будет равно 4 и матрица SQ может быть представлена, в матричном виде как квадратная матрица размером N*N = 4*4. В дальнейшем для упрощения понятия принципа работы матричного алгоритма в матрицу SQ запишем числа от 0 до 15 по порядку, хотя их порядок может быть изменён - по согласованию, между отправителем и получателем информации. Так как n=4 то все вычисления будут производиться по модулю $2^n = 2^4 = 16$.

С учётом этого на первом этапе вычислений матриц SQ может быть представлена, в следующем виде:

$$SQ=\begin{bmatrix} 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \\ 12 & 13 & 14 & 15 \end{bmatrix}$$

Так как вспомогательная матрица KQ вычисляется по формуле(2), то есть:

$$KQ = L*M \bmod 2^n ,$$

то заметим, что вычисление значений матрицы KQ производится с применением двух матриц- L (N*1) и M(1*N), и числа, входящие в данные матрицы могут выбираться произвольным образом из чисел, входящих в матрицу SQ, как было указано ранее. Для простоты и наглядности вычислений примем, что матрица M равна транспонированной матрице L, а в качестве произвольных чисел выберем значения 1,2,3,4.

Тогда матрица KQ вычисляется по модулю 16 следующим образом

$$KQ = L * L^T = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} * (1\ 2\ 3\ 4) = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 6 & 8 \\ 3 & 6 & 9 & 12 \\ 4 & 8 & 12 & 0 \end{bmatrix} \bmod 16$$

Значение матрицы SQL вычисляется в соответствии с уравнением (3) - суммируя матрицы SQ и KQ по модулю16, получим

$$SQL = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 4 & 5 & 6 & 7 \\ 8 & 9 & 10 & 11 \\ 12 & 13 & 14 & 15 \end{bmatrix} + \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 6 & 8 \\ 3 & 6 & 9 & 12 \\ 4 & 8 & 12 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 3 & 5 & 7 \\ 6 & 9 & 12 & 15 \\ 11 & 15 & 3 & 7 \\ 0 & 5 & 10 & 15 \end{bmatrix} \bmod 16$$

Выборка чисел для применения в поточном шифровании в качестве гаммы может быть произведена любым способом, как по порядку (первая строка, вторая и т.д.), так и начиная с последнего элемента данной таблицы после перевода значений выбранных чисел из десятичной в двоичную систему счисления,

Данный матричный алгоритм даёт возможность применения различных вариантов для дальнейших вычислений матрицы SQL и её составляющих, таких как матрица SQ, и матрица KQ.

Например, можно получить новое значение матрицы SQL как результат произведения матриц SQ и KQ, то есть

$$SQL = SQ * KQ, \quad (4)$$

учитывая, что все вычисления производятся по модулю 16, как в рассмотренном выше примере.

Применительно к матрице KQ для получение новых значений также возможны следующие варианты:

во - первых, возможно преобразовать матрицу KQ с помощью выбора новых чисел (произвольных) в матрицах L, M или, например, можно выбрать в качестве матрицы L- i-ый столбец матрицы SQ, а в качестве матрицы M- j-ую строку матрицы SQ;

во – вторых, сдвинув «старые» элементы матриц L, M на определённое число позиций (одинаковое или разное-выбранное по заранее согласованному алгоритму) и получить в результате их перемножения новое значение матрицы KQ;

в – третьих, можно ввести новый элемент – число μ, где μ - скалярная величина, находящаяся в диапазоне $1 < \mu \leq 2^n-1$ и умножить матрицу KQ, полученную на предыдущем этапе вычислений на μ- в результате чего все элементы матрицы KQ будут умножены на данный коэффициент и после применения к ним операции вычисления по модулю 2^n, в результате будет получено новое значение матрицы KQ на новом этапе вычислений.

После этих преобразований учитывая полученное обновлённое значение матрицы KQ получим новое значение матрицы SQL, элементы которой, то есть числа SQL_{ij} в двоичной форме записи могут быть использованы как числа k_i для работы алгоритма поточного шифрования.

Таким образом, предложенный матричный алгоритм может быть применён для получения последовательности псевдослучайных чисел(гамма) и в поточном шифровании.

Библиографический список:

1. Рябко Б.Я., Фионов А.Н. Криптография в информационном мире. - М.: Горячая линия-Телеком, 2018. -300с.:ил.