УДК 681.084

АЛГОРИТМ ФАКТОРИЗАЦИИ

к.т.н. В.Я. Певнев, И.В. Дулгер
(представил д.т.н. проф. Ю.В. Стасев)

В статье представлены алгоритмы факторизации больших чисел, в которых операции умножения заменены операциями сложения.

В настоящее время для защиты информации используются или находятся на стадии разработки различные блочные криптоалгоритмы. К ним относятся алгоритмы DSA и RSA [1, 2, 3] (несимметричные), а также стандарт DES [4] и стандарт ГОСТ 28147-89 [5] (симметричные). Основным преимуществом блочных алгоритмов является их приспособленность и удобство использования в системах и сетях связи. Это обусловлено тем, что в блочных алгоритмах каждый блок данных шифруется одним и тем же ключом. При этом необходимо обеспечить синхронизацию только в пределах границ блока.

Использование несимметричных (ключи шифрования и дешифрования различны) криптоалгоритмов при выборе соответствующих параметров позволяет обеспечить доказанную вычислительную стойкость, т.е. количество переборов, необходимых для расшифровки настолько велико, что осуществить их не представляется возможным, либо информация потеряет актуальность к моменту ее расшифровки.

Стойкость несимметричной двухключевой системы зависит от сложности решения задачи разложения числа **N** на простые сомножители **P** и **Q**. Такая система является доказано стойкой, т.к. доказав стойкость числа **N** к разложению можно доказать и стойкость самой системы.

Сама задача факторизации относится к классу **NP** - полных [1]. Это означает, что сложность ее решения растет по экспоненте от размера входа.

Существует большое количество алгоритмов разложения числа на простые сомножители. Это алгоритм Эвклида, Ленстры, Поларда и др. Все они эффективны до определенных размеров исходного числа.

Исходя из требований к простым сомножителям **P** и **Q** в работе предполагается алгоритм, позволяющий уменьшить временные затраты при поиске этих чисел.

Для раскрытия сути алгоритма введем следующие понятия:

N - число, которое необходимо факторизовать;
$\mathbf{P_i}$, $\mathbf{Q_i}$ - переменные, которые зависят от шага алгоритма;
$\mathbf{T_i}$ - произведение чисел $\mathbf{P_i}$ и $\mathbf{Q_i}$;
i - шаг алгоритма.

На первом шаге алгоритма переменным **P** и **Q** присваиваются значения, которые равны соответственно:

$$P_1 = \lfloor \sqrt{N} \rfloor,$$

$$Q_1 = \lceil \sqrt{N} \rceil = P_1 + 1.$$

Введем переменную Δ_i, которая представляет собой разность между числом **N** и произведением переменных **P** и **Q** на **i** - м шаге алгоритма:

$$\Delta_i = N - P_i \bullet Q_i \text{ i},$$

где $\begin{cases} P_i < P_{i-1} \\ Q_i > Q_{i-1} \end{cases}$.

Очевидно, что Δ_i может принимать как положительное значение, так и отрицательное, а также может равняться **0**. В случае, если Δ_i равно **0**, то P_i и Q_i – искомые, простые числа. Если Δ_i не равно **0**, необходимо изменить P_i и Q_i.

Если $\Delta_i < 0$, то необходимо уменьшить P_i на **1**. Если $\Delta_i > 1$, то необходимо увеличить Q_i на **1**.

Рассмотрим случай, когда $\Delta_i < 0$. В этом случае имеем:

$$\Delta_i = N - P_i \bullet Q_i < 0,$$

тогда для получения Δ_{i+1} необходимо:

$$P_{i+1} = P_i - 1, \quad (1)$$

$$\Delta_{i+1} = N - (P_i - 1) \bullet Q_i = N - P_i \bullet Q_i + Q_i = \Delta_i + Q_i. \quad (2)$$

следовательно, $\Delta_{i+1} = \Delta_i + Q_i$. Это соотношение будет всегда положительным, т.к. $-P_i > \Delta_i > 0$ и всегда $P_i < Q_i$.

В случае, когда $\Delta_i > 0$ имеем:

$$\Delta_i = N - P_i \bullet Q_i > 0,$$

тогда для получения Δ_{i+1} необходимо:

$$Q_{i+1} = Q_i + 1, \quad (3)$$

$$\Delta_{i+1} = N - P_i \bullet (Q_i + 1) = N - P_i \bullet Q_i - P_i = \Delta_i - P_i. \quad (4)$$

Итак, мы получили $\Delta_{i+1} = \Delta_i - P_i$. Полученное соотношение может принимать любое значение (как положительное, так и отрицательное), т.к. $0 < \Delta_i < Q_i$.

Таким образом, для получения Δ_{i+1} необходимо проделать следующую последовательность действий:

$$\begin{cases} P_{i+1} = P_i - 1 \text{ при } \Delta_i < 0 \\ Q_{i+1} = Q_i \\ Q_{i+1} = Q_i + 1 \text{ при } \Delta_i > 0 \\ P_{i+1} = P_i \end{cases} \quad (5)$$

$$T_{i+1} = P_{i+1} \bullet Q_{i+1}; \quad (6)$$

$$\Delta_{i+1} = N - T_{i+1}.$$

Но, исходя из формул (2) и (4), указанную выше последовательность можно записать следующим образом:

$$\begin{cases} \Delta_{i+1} = \Delta_i + Q_i \text{ при } \Delta_i < 0 \\ P_{i+1} = P_i - 1 \\ \Delta_{i+1} = \Delta_i - P_i \text{ при } \Delta_i > 0 \\ Q_{i+1} = Q_i + 1 \end{cases} \quad (7)$$

Получив Δ_{i+1}, сравниваем его с **0** и, если Δ_{i+1} не равен **0**, определяем знак и делаем следующий шаг. В конце концов, мы получим $\Delta_i = 0$ и найдем искомые **P** и **Q**.

Достоинством изложенного выше алгоритма является быстродействие, т.к. при его реализации операции сложения - вычитания (5), умножения больших чисел (6) заменяются операциями сложения - вычитания (7). Недостатком данного алгоритма является то обстоятельство, что в качестве возможных **P** и **Q** рассматриваются как четные так и нечетные числа, а по условию задачи **P** и **Q** простые, и следовательно нечетные числа.

Попытаемся устранить этот недостаток:

Введем переменные

$$P'_1 = \lfloor \sqrt{N} \rfloor,$$

$$Q'_1 = \lceil \sqrt{N} \rceil = P'_1 + 1.$$

В том случае, если P'_1 четное число, то примем $P_1 = P'_1 - 1$, иначе $P_1 = P'_1$. Рассуждая аналогичным образом, для Q'_1 получим $Q_1 = Q'_1 + 1$, иначе $Q_1 = Q'_1$.

Теперь мы точно знаем, что P_1 и Q_1 нечетные числа и можем модифицировать формулы (1, 2) и (3, 4) следующим образом:

$P_{i+1} = P_i - 2$,

$\Delta_{i+1} = N - (P_i - 2) \bullet Q_i = N - P_i \bullet Q_i + 2 \bullet Q_i = \Delta_i + 2 \bullet Q_i$

$Q_{i+1} = Q_i + 2$,

$\Delta_{i+1} = N - P_i \bullet (Q_i + 2) = N - P_i \bullet Q_i - 2 \bullet P_i = \Delta_i - 2 \bullet P_i$.

Теперь полученный алгоритм будет рассматривать в качестве возможных **P** и **Q** только нечетные числа, и количество итераций уменьшится вдвое.

В представленной работе предлагаются два алгоритма факторизации. Достоинством этих алгоритмов является замена операций умножения на операции сложения.

ЛИТЕРАТУРА

1. Феллер В. Введение в теорию вероятностей и ее приложения. – М.: Мир, 1964. – 498 с.

2. Горбенко И.Д., Долгов В.И., Потий А.В., Федорченко В.И. Анализ каналов уязвимости системы RSA // Безопасность информации, 1995, №8. - С. 22–26.

3. Диффи У. Первые десять лет криптографии с открытым ключом // ТИИЭР, т.76, №5, 1988. С. 54 - 74.

4. Пярин В.А. Генерация, распределение и использование криптографических ключей // Защита информации. – 1994. - №1.- С. 157 - 184.

5. Ивашкин А.В. Методика испытаний и экспериментальных исследований УП . Отчет о ОКР, ХГТУРЭ, 1996. – 87 с.

6. Эльтанов Б.А. Развитие метода решета. М.: Статистика, 1986. – 157 с.