Реализованные угрозы с их стороны и негативные информационные воздействия на социальные объекты и группы могут привести к возникновению серьезных критических ситуаций в национальном масштабе.

3. Для своевременного выявления этих угроз и негативных информационных воздействий и их нейтрализации необходима разработка и реализация комплекса организационных и научно-технических мероприятий по противодействию. В основе этих мероприятий лежит создание единой системы радиочастотного и сетевого мониторинга, объединенного со СППР.

4. Основной функцией данной интеллектуальной системы будет выявление по ряду косвенных и прямых признаков возможных угроз и негативных воздействий на телекоммуникационные сети и социальные объекты, а также выработка мер по их своевременной нейтрализации.

5. Для реализации этих возможностей СППР должна иметь определенную структуру и возможности. В данной работе определены структура и основные требования к системе поддержки принятия решений и ее основным элементам. В частности, в подсистеме радиочастотного мониторинга должна быть обеспечена возможность распознавания и анализа сигналов и их декодирование. В самой СППР должна быть обеспечена возможность анализа и интерпретации текстовых сообщений для выявления негативных информационных воздействий.

Литература: ***1.*** *Семенюченко А. В. IPS – современные технологии обнаружения и предотвращения атак // «IT спец», журнал для IT – профессионалов. – 2007.-№5. – с.46 – 55.* ***2.*** *Зайцев О. Н. Oracle – безопасность сервера // «IT спец», журнал для IT – профессионалов. – 2007.-№11. – с.46 – 50.* ***3.*** *Джексон П. Введение в экспертные системы. Пер с англ.: Учебное пособие – М.: Издательский дом «Вильямс», 2001. – 624 с.* ***4.*** *Информационно-вычислительные системы принятия решений / В. В. Хаджинов, В. А. Быков, И. А. Храмова, В. Г. Усачев – Киев: Наукова думка, 1992. – 140 с.* ***5.*** *Человеко-машинные системы принятия решений с элементами искусственного интеллекта / Б. М. Герасимов, В. А. Тарасов, И. В. Токарев – Киев: Наукова думка, 1993.– 184 с.* ***6.*** *Горелик А. Л., Скрипкин В. А. Методы распознавания. Уч. пособие для вузов. – М.: Высшая школа, 1977. – 222 с.* ***7.*** *Хореев А. А. Теоретические основы оценки возможностей технических средств разведки: Монография. М.: МО РФ, 2000. – 255 с.*

УДК 511.215

РАЗРЯДНЫЙ ПОДХОД К ФАКТОРИЗАЦИИ ЧИСЛА – ПРОИЗВЕДЕНИЯ ДВУХ ПРОСТЫХ ЧИСЕЛ

Артём Жилин

Институт специальной связи и защиты информации НТУУ «КПИ»

Анотація: **Розглянуто розрядний підхід до факторизації чисел, який за допомогою функції nat(a) зводиться до рішення системи рівнянь, яка має вид поліномів Жегалкіна. Для рішення системи рівнянь запропоновано «ваговий» підхід із нормалізацією системи рівнянь.**

Summary: **The bit going is considered near factorization of numbers, which is taken by the entered function of nat(a) to the decision of the system of equalizations, having the appearance of Zhegalkin's polynomials. For the decision of the system of equalizations «gravimetric» approach is offered with normalization of the system of equalizations.**

Ключевые слова: **Факторизация, система булевых уравнений, «весовой» подход.**

I Введение

Концепция асимметричных криптосистем (криптографии с открытым ключом) была предложена Уитфилдом Диффи и Мартином Хеллманом [1]. Основным пунктом концепции было предложение использовать ключи парами, состоящими из ключа зашифрования и ключа расшифрования, которые невозможно вычислить один из другого.

Как правило, алгоритмы, используемые для реализации асимметричных криптосистем, основываются на решении одной из трудных математических задач. Классическим примером такой задачи является факторизация (разложение на простые множители) большого числа. В частности, на этой задаче основан один из самых известных криптоалгоритмов - RSA.

II Основная часть

Считается [1], что сложность атаки на криптоалгоритм асимметричной криптосистемы определяется сложностью алгоритма решения задачи факторизации большого числа, являющегося произведением двух простых чисел.

Наиболее тривиальным алгоритмом факторизации чисел является полный перебор возможных делителей. Но сложность этого алгоритма очень большая и оценивается как $O(N^{1/2})$, где N - число, подлежащее факторизации. Разработаны также и другие алгоритмы факторизации, которые условно можно разделить на две группы. Первая группа — экспоненциальные алгоритмы, сложность которых экспоненциально зависит от длины входящих параметров (то есть от длины самого числа в бинарном представлении). Вторая группа — субэкспоненциальные алгоритмы. К первой группе относятся: ρ-алгоритм Полларда, p-1 алгоритм Полларда, p+1 алгоритм Вильямса, метод квадратичных форм Шенкса, метод Шермана — Лемана. Субэкспоненциальные алгоритмы: метод непрерывных дробей, метод квадратичного решета, метод эллиптических кривых. В настоящее время самыми эффективными алгоритмами факторизации являются: специальный метод решета числового поля и общий метод решета числового поля, которые характеризуются сложностью порядка $L_N(1/3,(64/9)^{1/3})$, при $L_N(\alpha,c)=O(\exp((c+o(1))(\log N)^{\alpha}(\log\log N)^{1-\alpha}))$, где N — число, подлежащее факторизации, $0 < \alpha < 1$ и c — некоторые константы. Проведенный анализ показал, что все существующие методы факторизации основываются на нахождении НОД с ограничением области перебора возможных значений делителей или на модификации метода решета числового поля. Т. е. они оперируют свойствами простых целых чисел, используя как обычную, так и модулярную алгебру. Вместе с тем отмечается [1], что учет особенностей структуры сомножителей может повысить эффективность алгоритмов и снизить показатели сложности.

Представленный в работе [2] метод факторизации, основанный на разрядном подходе, предполагает работу со структурой числа, которое представлено в двоичной форме, и базируется на единственности разложения числа на простые сомножители и правилах умножения чисел в двоичной системе счисления.

Факторизуемое число в двоичной форме будем представлять в следующем виде: $Z = z_0 z_1 z_2 z_3 z_4 ... z_n$, где z_0 - старший разряд числа, z_n - младший разряд числа. Полагаем, что Z является произведением двух простых чисел $X = x_1 x_2 x_3 x_4 ... x_k$ и $Y = y_1 y_2 y_3 y_4 ... y_k$, где x_1, y_1 - старшие разряды, x_k, y_k - младшие разряды чисел X и Y соответственно:

$$Z = X * Y .$$
$$z_i, x_j, y_j \in \{0,1\}, \quad i = 0...n, \quad j = 1...k, \quad k = (n+1)/2$$

Умножение чисел в двоичной системе счисления подчиняется правилам умножения чисел «в столбик». Для любых одноразрядных значений действуют следующие правила:

$x_i * x_i = x_i$

$x_i * 1 = x_i$

$x_i * 0 = 0$

$x_i + x_i = 0$ с переносом x_i в старший разряд.

$x_i + x_i + x_i = x_i$ с переносом x_i в старший разряд.

Умножая числа X и Y на основе этих правил в символьном виде, получаем следующую систему уравнений:

$$\begin{gathered} z_0 = P_1 \\ z_1 = y_1 x_1 + P_2 \\ z_2 = y_1 x_2 + y_2 x_1 + P_3 \\ z_3 = y_1 x_3 + y_2 x_2 + y_3 x_1 + P_4 \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ z_{l+1} = y_1 x_k + y_2 x_{k-1} + \cdots + y_k x_1 + P_i \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \end{gathered} \qquad (1)$$

$$z_{n-2} = y_{k-2}x_k + y_{k-1}x_{k-1} + y_k x_{k-2} + P_{n-1}$$
$$z_{n-1} = y_{k-1}x_k + y_k x_{k-1}$$
$$z_n = y_k x_k$$

где P_n - разрядный перенос.

Определим, что система в заданном виде будет иметь старшие и младшие разрядные уравнения. Отношение старшинства между уравнениями определяется в соответствии с тем, к какому разряду факторизуемого числа оно относится. Решением системы уравнений будут числа $x_j, y_j \in \{0,1\}, j = 1...k$, которые обращают уравнения в тождества при заданных значениях z_n.

Данная система уравнений имеет вид системы из *N* (где *N* – разрядность факторизуемого числа), в общем случае, нелинейных алгебраических уравнений с *N* неизвестными, имеющими характер булевых переменных. При этом сами нелинейности имеют характер произведений неизвестных. Следует отметить, что получаемая система уравнений является не совсем обычной. Ее «необычность» проявляется, во-первых, в том, что, в зависимости от структуры факторизуемого числа, между уравнениями могут существовать дополнительные связи, обусловленные наличием переносов из младших разрядов в старшие. Очевидно, что характер таких связей очень сильно зависит от структуры сомножителей, отображенных в их произведении. Именно по этой причине положение каждого уравнения в системе должно быть строго зафиксированным и не может быть изменено. Т.е. решение системы уравнений не является инвариантным к перестановке строк ее матрицы, более того, строки являются взаимосвязанными, начиная с последней (младшей, нижней). Во-вторых, количество нелинейных членов в каждом уравнении системы и степень самой нелинейности так же существенно зависят от структуры факторизуемого числа. Методы решения такой задачи в общем виде к настоящему времени не были известны, за исключением, пожалуй, методов перебора.

Таким образом, решение задачи факторизации с использованием разрядного подхода предполагает преодоление двух трудностей: построение формального механизма учета переносов в общем случае и решение системы нелинейных булевых уравнений специальной формы.

В качестве формального механизма учета переносов и представления системы уравнений в несвязанном виде вводится дискретная функция натурального аналитического трансфера - *nat(a)*, которая возвращает количество единиц переноса в зависимости от значения, которое принимает низшее разрядное уравнение. Ниже приведена таблица значений, которые принимает функция *nat(a)* в зависимости от значений входящих параметров.

Таблица 1 – Значение функции *nat(a)* в зависимости от значений входящих параметров

a	0	1	2	3	4	5	6	7	…	2*i	2*i+1
nat(a)	0	0	1	1	2	2	3	3	…	i	i

В общем случае аналитическое выражение функции *nat(a)* имеет вид:

$$nat(a_1 + a_2 + .. + a_n) = \sum_{i=1}^{n} nat(a_i) + \sum_{i=1}^{n-1}(a_i * \bigoplus_{j=i+1}^{n} a_j) \qquad (2)$$

Приведенная функция обладает следующим свойством:

$$nat(a) = 0,\ a = \{0,1\}$$

Основываясь на данном свойстве, можно утверждать, что при использовании функции *nat(a)* возможно сокращение первого слагаемого $\sum_{i=1}^{n} nat(a_i)$, при условии, что выражением под функцией являются или данные неизвестные, или их произведение, или их сума по модулю два.

Для того, что бы представить систему уравнений (1) в несвязанном виде и учесть переносы между разрядными уравнениями, каждое разрядное уравнение начиная с младшего (нижнего) берется по функции *nat(a)* и результат подставляется в старшее, по отношении с выбранным, уравнение. Уравнение, с которым проводилась операция, берется по модулю два, так как разрядные связи уже учтены, и оно будет равно левой части. С *n-го* по *(n*+1*)/*2 уравнения перед каждым применением функции необходимо выполнять выражение неизвестной *ym*, где *m*=1…(*n*+1)/2-1, через неизвестные разряды числа *X* и левую часть. При этом выраженная неизвестная берется по модулю два и, после упрощения, подставляется во всю систему уравнений.

При использовании функции *nat(a)* система с разрядными связями преобразовалась в систему нелинейных алгебраических уравнений без таких связей. При этом сами уравнения имеют вид сумм некоторых вариантов произведений неизвестных по модулю два (т. е. уравнения получаемой системы имеют вид полиномов Жегалкина).

$$
\begin{aligned}
z_0 &= \bigoplus_{i0_1,\ldots,i0_{k-1}} a_{i0_1,\ldots,i_{0k}-1} x_{i0_1} \ldots x_{i0_{k-1}}, a_{i0_1,\ldots,i0_{k-1}} \in \{0,1\} \\
z_1 &= \bigoplus_{i1_1,\ldots,i1_{k-1}} a_{i1_1,\ldots,i1_k-1} x_{i1_1} \ldots x_{i1_{k-1}}, a_{i1_1,\ldots,i1_{k-1}} \in \{0,1\} \\
z_2 &= \bigoplus_{i2_1,\ldots,i2_{k-1}} a_{i2_1,\ldots,i2_k-1} x_{i2_1} \ldots x_{i2_{k-1}}, a_{i2_1,\ldots,i2_{k-1}} \in \{0,1\} \\
z_3 &= \bigoplus_{i3_1,\ldots,i3_{k-1}} a_{i3_1,\ldots,i3_k-1} x_{i3_1} \ldots x_{i3_{k-1}}, a_{i3_1,\ldots,i3_{k-1}} \in \{0,1\} \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
z_l &= \bigoplus_{il_1,\ldots,il_{k-1}} a_{il_1,\ldots,il_k-1} x_{il_1} \ldots x_{il_{k-1}}, a_{il_1,\ldots,il_{k-1}} \in \{0,1\} \\
z_{l+1} &= z_{l+1} \\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
z_{n-2} &= z_{n-2} \\
z_{n-1} &= z_{n-1} \\
z_n &= 1
\end{aligned}
\qquad (3)
$$

Таким способом удается преодолеть первую трудность.

С целью решения получаемой системы нелинейных уравнений предлагается подход, основанный на учете «весов» каждой переменной в системе. Опишем суть данного подхода.

Для удобства реализации подхода и наглядности строится таблица, которая приведена ниже.

Таблица 2 – Выбор доминирующей неизвестной

	x_1	x_2	…	x_k	Левая часть
0	x_{1z0}	x_{2z0}	…	x_{kz0}	*z0*
1	x_{1z1}	x_{2z1}	…	x_{kz1}	*z1*
2	x_{1z2}	x_{2z2}	…	x_{kz2}	*z2*
…	…	…	…	…	…
l	x_{1zl}	x_{2zl}	…	x_{kzl}	*zl*
Всего	по x_1	по x_2	…	по x_k	

В данной таблице пронумерованным строкам соответствуют уравнения системы, а столбцам – обозначенные неизвестные. Последнему столбцу присваивается значение левой части уравнений. Нижняя же строка является суммой соответствующей неизвестной во всей системе уравнений. Получается, что соответствующая ячейка будет являться количественным показателем неизвестной в определенном уравнении.

Для последовательного определения значения неизвестной выбирается доминирующая неизвестная путем подсчета количества вхождений каждой из неизвестных во все уравнения системы. Доминирующей неизвестной присваивается то значение, по которому она доминирует относительно левой части системы. Согласно же таблице 2 выбирается та переменная, которая имеет максимальное значение в последней строке. После этого для выбранной неизвестной подсчитывается ее количественный показатель относительно значений правого столбца (относительно 0 и 1). По какому из значений количество неизвестных будет больше, то и будет значением неизвестной. Полученное значение подставляется в систему, в результате чего данная переменная исключается из системы, а система упрощается. В получаемой после этого системе уравнений снова выбирается доминирующий элемент, и снова подставляется значение доминирования. Этот процесс продолжается до тех пор, пока система не приводится к виду, допускающему тривиальное решение.

Следует отметить, что при реализации изложенного подхода необходимо выполнять ряд дополнительных преобразований. Эти преобразования направлены на устранение избыточности количества членов уравнений

системы. Необходимость данных действий вызвана тем, что приведенный подход основан на подсчете неизвестных в системе, а когда существует избыточность членов уравнений, то она влияет на количество неизвестных в системе, но не влияет на общую зависимость в структуре системы. Чтобы минимизировать этот недостаток необходимо провести процедуру нормализации системы, которая заключается в суммировании по модулю два уравнений системы. Также существует второстепенные преобразования, относительно которых выработаны рекомендации по их использованию.

Анализ приведенного подхода показывает, что его вычислительная сложность ниже экспоненциальной, так как самая трудоемкая часть алгоритма (нормализация системы) имеет максимальное количество операций, которое равно сочетанию из $N/2$ по 2, где N – разрядность числа, подлежащего факторизации.

III Заключение

В настоящее время еще нельзя утверждать, что приведенный подход является алгоритмом в том смысле, что известны конкретные действия для любой комбинации значений разрядов в сомножителях. Но уже понятно, что на основе данного подхода может быть организована атака на ассиметричные криптосистемы, сложность реализации которой может быть ниже экспоненциальной. В настоящее время ведется дальнейшая работа по развитию данного подхода, а также всех процедур, в него входящих.

*Литература: **1**. Шнайер Б. Прикладная криптография. – М.: Издательство ТРИУМФ, 2003 – 816 с.: ил. **2**. Жилин А. В., Мохор В. В. Структурный метод факторизации больших целых чисел// Моделювання та інформаційні технології. Зб. наук. пр. ІПМЕ НАН України, спец. випуск - К.: 2008. - С. 49 – 57.*

УДК 004.4

ВОЗМОЖНОСТИ ИСПОЛЬЗОВАНИЯ СИСТЕМ ЗАЩИТЫ ИНФОРМАЦИИ В ERP-СИСТЕМАХ

Александр Голяка
Национальный банк Украины

Аннотация: **В работе рассматривается проблема защиты информации в ERP-системах, возможности ее решения с помощью внутренних механизмов систем на примере SAP R/3. Показана необходимость создания внешних криптографических библиотек защиты обязательно реализованных с использованием интерфейсов SSF и GSS API, которые открыты для внешних разработчиков в системах SAP. Описаны принципы работы функций SSF-API, форматы входных/выходных данных.**
Summary: **The defense of information in ERP-sistems is considered.**
Ключевые слова: **Защита информации в ERP-системах, форматы входных – выходных данных.**

Системы класса ERP – (Enterprise Resource Planning – Планирование Ресурсов Предприятия) – это комплекс интегрированных приложений, позволяющих создать единую среду для автоматизации планирования, учета, контроля и анализа всех основных бизнес-процессов предприятия. Такие системы предполагают управление всеми (или отдельными) ресурсами предприятия – трудовыми, финансовыми, материальными и пр. Как правило, ERP-система состоит из набора подсистем, таких как: финансы, снабжение и сбыт, хранение, производство и пр. Одной из наиболее совершенных и типичных представителей класса ERP-систем, являющаяся, по сути, отраслевым стандартом в этой области, является система SAP R/3, разработанная немецкой компанией SAP AG.

Система SAP R/3 основана на трехзвенной архитектуре. С точки зрения SAP R/3, трехзвенная архитектура состоит из презентационного уровня, уровня приложений и уровня базы данных (БД) (рис. 1).

С точки зрения аппаратных средств эти уровни независимо функционируют на разных компьютерах или совместно на одном в зависимости от конфигурации системы (одноуровневая/двухуровневая/трехуровневая). Кроме того, SAP R/3 позволяет распределять презентационный уровень и уровень приложений по нескольким компьютерам.

Для больших предприятий с точки зрения производительности наиболее эффективна трехуровневая конфигурация. В такой конфигурации несколько разных серверов приложений могут одновременно обрабатывать данные, хранящиеся на общем сервере БД.

Фактические вычисления и оценки системы выполняются на серверах приложений. Уровень БД представляет собой реляционную систему управления базой данных (РСУБД). Обмен данными между РСУБД и процессами приложений осуществляется через интерфейс SQL. Презентационный уровень