Новые и Модернизированные Алгоритмы для Системы RSA

Анатолий Агафонов, Анатолий Балабанов, Бартоломеу Изворяну, Ирина Кожухарь
Технический Университет Молдовы
vonofaga@mail.ru, balsoft@mcc.md, izvor@mail.utm.md, irina_cojuhari@ati.utm.md

Аннотация **— Одной из часто применяемых на практике асимметричных криптографических систем является система RSA. Среди ее рабочих задач и алгоритмов выделяется процедура генерации криптоключей. В работе авторов предложен усовершенствованный алгоритм определения простоты числа и, построенный на его базе, быстродействующий алгоритм генерации простых чисел.**

Ключевые слова**— криптография, криптосистема RSA, тесты простоты числа, генераторы криптоключей, скорость алгоритма, операционная емкость алгоритма, теория чисел, сравнительный анализ, сопоставительный анализ, формантный анализ, форманта числа.**

1. ВВЕДЕНИЕ

В настоящее время RSA является чаще всего используемой асимметричной криптосистемой с открытым (public) ключом, обеспечивающей такие механизмы защиты как шифрование и цифровую подпись [2, 4, 7, 10]. Алгоритм RSA, предложенный тремя исследователями-математиками Рональдом Ривестом (R. Rivest), Ади Шамиром (A. Shamir) и Леонардом Адльманом (L. Adleman) в 1977-78 годах, является одним из самых надежных и зачастую называется стандартом де факто в практике современной криптографии. Вне зависимости от официальных стандартов существование такого стандарта чрезвычайно важно для развития электронной коммерции и вообще экономики.

I. СОВРЕМЕННОЕ ЗНАЧЕНИЕ СИСТЕМЫ RSA

Единая система документооборота с открытым ключом предполагает использование электронно-цифровой подписи между пользователями различных государств, использующими разнообразное программное обеспечение на различных платформах. Распространение системы RSA дошло до такой степени, что ее учитывают и при создании новых стандартов. При разработке стандартов цифровых подписей в 1997 был разработан стандарт ANSI X9.30, поддерживающий Digital Signature Standard (стандарт Цифровой подписи). Годом позже был введен ANSI X9.31, в котором сделан акцент на цифровых подписях RSA, что отвечает фактически сложившейся ситуации, в частности для финансовых учреждений.

Любое ассиметричное шифрование предполагает наличие пары ключей, а именно открытого и закрытого. Алгоритм RSA основывается на использовании односторонних функций шифрования, обладающих известными свойствами:

- если x известно, то $y = f(x)$ вычислить относительно просто;
- если $y = f(x)$ известно, то вычислить x практически невозможно.

Основу криптографической системы с открытым ключом RSA составляют практические трудности решения задачи о разложении чисел на множители (задача факторизации), которая является вычислительно однонаправленной задачей.

Алгоритм создания открытого и закрытого ключей:

1. Выбираются 2 простых числа p и q.
2. Вычисляется их произведение $N=p*q$.
3. Вычисляется значение функции Эйлера $f(p,q) = (p-1)*(q-1)$.
4. Выбирается простое число e, взаимно простое с $f(p,q)$.
5. Выбирается число d, удовлетворяющее условию $e*d \bmod f(p,q)=1$.
6. Пара $[e,N]$ – открытый ключ шифрования.
7. Пара $[d,N]$ – закрытый ключ шифрования.

Преимущества системы RSA:

- Преимущество асимметричных шифров перед симметричными состоит в отсутствии необходимости предварительной передачи секретного ключа по надёжному каналу.
- В симметричной криптографии – только один ключ и он держится в секрете обеими сторонами, а в асимметричной криптосистеме – два ключа, но только один ключ - секретный, и он известен только одной стороне.
- При симметричном шифровании для повышения надежности (криптостойкости) необходимо обновлять ключ после каждого сеанса передачи шифрованной информации, тогда как в асимметричных криптосистемах пару (e,d) можно не менять достаточно длительное время.
- В больших сетях число ключей в асимметричной криптосистеме значительно меньше, чем в симметричной.
- Асимметрично-ключевая криптография успешно работает для установления подлинности цифровых подписей и при рассылке криптоключей.

Недостатки системы RSA:

- Преимущество алгоритма симметричного шифрования перед асимметричным заключается в том, что в первый относительно легко внести изменения.
- Хотя сообщения надежно шифруются, но «засвечиваются» получатель и отправитель самим фактом пересылки шифрованного сообщения.
- Асимметричные алгоритмы используют более длинные ключи, чем симметричные.
- Алгоритм RSA намного медленнее, чем DES и другие алгоритмы блокового шифрования. Так, процесс шифрования-расшифрования с использованием пары ключей проходит на два-три порядка медленнее, чем шифрование-расшифрование того же текста симметричным алгоритмом. При этом шифрование данных идет быстрее, чем расшифровка, а проверка подписи – быстрее, чем подписание. Из-за особенностей алгоритмов, лежащих в основе систем с открытым ключом, быстродействие процедур обработки единичного блока информации обычно в десятки раз меньше, чем быстродействие систем с симметричным ключом на блоке той же длины. Для повышения эффективности систем с открытым ключом часто применяются смешанные методы, реализующие криптографические алгоритмы обоих типов.
- В чистом виде асимметричные криптосистемы требуют существенно бóльших вычислительных ресурсов, поэтому на практике их используют в сочетании с другими алгоритмами.
  1. Для электронной цифровой подписи (ЭЦП) сообщение предварительно подвергается хешированию, а с помощью асимметричного ключа подписывается лишь относительно небольшой результат хеш-функции.
  2. Асимметричные криптосистемы используются в форме гибридных криптосистем, где большие объёмы данных шифруются симметричным шифром на сеансовом ключе, а с помощью асимметричного шифра передаётся только сам сеансовый ключ.

1.1. ИСПОЛЬЗОВАНИЕ КРИПТОСИСТЕМЫ RSA В НАСТОЯЩЕЕ ВРЕМЯ

Криптосистема RSA используется в самых различных программных продуктах, на различных платформах и во многих отраслях [2, 4, 7, 10]. В настоящее время RSA встраивается во многие коммерческие продукты, число которых постоянно увеличивается. Также ее используют операционные системы Microsoft, Apple, Sun и Novell. В аппаратном исполнении RSA-алгоритм применяется в защищенных телефонах, на сетевых платах Ethernet, на смарт-картах и широко используется в криптографическом оборудовании THALES (Racal). Кроме того, этот алгоритм входит в состав всех основных протоколов для защищенных коммуникаций Internet, в том числе S/MIME, SSL и S/WAN, а также используется во многих учреждениях, например, в правительственных службах, большинстве корпораций, государственных лабораториях и университетах.

Технологию шифрования RSA BSAFE используют свыше 500 миллионов пользователей всего мира. Так как в большинстве случаев при этом используется алгоритм RSA, то его можно считать наиболее распространенной в мире криптосистемой с открытым ключом и это количество имеет явную тенденцию к увеличению по мере роста сети Internet.

II. ТЕСТЫ ПРОСТОТЫ ЧИСЛА

2.1. КРАТКИЙ ОБЗОР СУЩЕСТВУЮЩИХ ВЕРОЯТНОСТНЫХ АЛГОРИТМОВ ОПРЕДЕЛЕНИЯ СОСТАВНОГО/ ПРОСТОГО ЧИСЛА.

Все существующие алгоритмы тестирования чисел на простоту можно разделить на два класса:

- так называемые детерминированные тесты, которые в результате дают гарантированно точный ответ простое ли исследуемое число или нет,
- вероятностные тесты, результат выполнения которых является достоверным лишь с некоторой достаточно высокой вероятностью.

В основе наиболее употребляемых вероятностных тестов простоты числа лежит Малая теорема Ферма, которая утверждает, что разность $(A^{N-1} - 1)$ всегда делится на N если N простое число и $A < N$, то есть:

$$A^{N-1} - 1 = N \cdot z. \quad (1)$$

Задача повышения достоверности вероятностных тестов стала актуальной для криптографии, после того как были обнаружены «числа Кармайкла» и стало ясно, что тест на основе Малой теоремы Ферма «даёт сбои». Хотя таких чисел и не много (всего–то 16 среди первых 100000 натуральных чисел). Но теоретики шифрования не чувствовали себя спокойными, пока не было найдено «противоядие» от этих «вредных чисел». Первый такой тест, лишённый недостатка теста Ферма, был предложен Леманном. Тест «реагировал» на числа Кармайкла и отсеивал их как составные, каковыми они на самом деле и являются.

Совершенствование вероятностных тестов шло по линии разложения левой части в уравнении Ферма (1) на как можно большее число сомножителей, поскольку количество сомножителей уменьшает время обнаружения составного числа [2, 4, 7, 9]. Дальнейшее развитие алгоритмов вероятностного тестирования простоты числа лучше всего проследить с помощью таблицы 1.

Пояснение к таблице 1:. N есть тестируемое на простоту число, ***A*** – **индикативное число**, с помощью которого данным тестом определяется тип числа ***N*****. Контрольным (критическим) числом *Ак* (*Акр*)** называется такое индикативное число A, при котором обнаруживается составная природа тестируемого числа. **Псевдопростое** число ***Nп*** данного алгоритма – это такое составное число N, которое при **данном** индикативном числе A, идентифицируется как простое. **Сильно псевдопростым** числом ***Nсп*** данного алгоритма называется такое составное число N, которое при **нескольких** индикативных числах A, идентифицируется как простое.

Таблица 1. Развитие тестов простоты

Математическая формула	Название теста	Особые числа (название)	Примеры особых чисел
1 $(A^{(N-1)} - 1 = Ny$	Ферма	Числа Кармайкла Псевдопростое	Nп: 341(A=2), Nп: 217(A=5) и т.д.
2 $(A^{0.5(N-1)} - 1)(A^{0.5(N-1)} + 1) = Ny$	Эйлера – Леманна	Псевдопростое Эйлера-Леманна	Nп: 341(A=2), Nп: 217(A=5) и т.д.
3 $A^{0.5(N-1)} \equiv (A/N) \bmod N$	Соловея - Штрассена	Псевдопростые Сильно псевдопростые	Nп: 341(A=2), Nп: 781(A=5) и т.д. Ncп: 1373653(A=2 и3), Ncп: 25326001(A=2,3 и 5) и т.д.
4 $(A^m - 1)(A^m + 1)(A^{2m} + 1)\ldots \equiv \bmod N$	Миллера -Рабина	Псевдопростые Сильно псевдопростые	Nп: 341(A=2), Nп: 781(A=5) и т.д. Ncп:1373653(A=2 и3) Ncп: 25326001(A=2,3 и 5) и т.д.

Конечным вероятностным тестом, ставшим наиболее употребительным в практике определения простоты числа, стал тест Миллера – Рабина (М-Р). В теоретических работах по данному тесту для определения простоты числа N рекомендуется использовать в качестве индикативных чисел любые, случайным образом выбранные, числа в диапазоне от 2 до N–1. Однако в повседневной практике закрепился такой подход использования теста Миллера - Рабина, когда берутся первые простые числа из этого ряда. Наши проверки чисел на простоту (тысячи экспериментов) показали его высокую эффективность уже с первой итерации ($Акр$=3) (разумеется, если проверяемое число не является сильно псевдопростым).

2.2. АЛГОРИТМ ГЕНЕРАЦИИ КРИПТОКЛЮЧЕЙ

Ключ — секретная информация, используемая криптографическим алгоритмом при шифровании/расшифровании сообщений, постановке и проверке цифровой подписи. При использовании одного и того же алгоритма результат шифрования зависит от ключа. Для современных весьма стойких алгоритмов криптографии утрата ключа приводит к практической невозможности расшифровать информацию.

Количество информации в ключе принято измерять в битах. Для современных симметричных алгоритмов основной характеристикой криптостойкости является длина ключа. Шифрование с ключами длиной 128 бит и выше считается *сильным*, так как для расшифровки информации без ключа требуются годы работы мощных суперкомпьютеров.

Для асимметричных алгоритмов, основанных на классических задачах теории чисел (в RSA это - проблема факторизации), в силу их особенностей, минимальная достаточно надёжная длина ключа в настоящее время -1024 бит.

Криптографические ключи различаются согласно алгоритмам, в которых они используются.

- **Симметричные ключи** — ключи, используемые в симметричных алгоритмах (шифрование, выработка кодов аутентичности). Главное свойство симметричных ключей: для выполнения как прямого, так и обратного криптографического преобразования (шифрование/ расшифровывание) необходимо использовать один и тот же ключ (либо же ключ для обратного преобразования легко вычисляется из ключа для прямого преобразования, и наоборот). С одной стороны, это обеспечивает более высокую конфиденциальность сообщений, с другой стороны, создаёт проблемы распространения ключей в системах с большим количеством пользователей.
- **Асимметричные ключи** - ключи, используемые в асимметричных алгоритмах; вообще говоря, являются **ключевой парой**, поскольку состоят из двух ключей:
  - **Закрытый ключ** (Private key) — ключ, известный только своему владельцу. Только сохранение пользователем в тайне своего закрытого ключа гарантирует невозможность подделки злоумышленником документа и цифровой подписи от имени заверяющего.
  - **Открытый ключ** (Public key) — ключ, который может быть опубликован и используется для проверки подлинности подписанного документа, а также для предупреждения мошенничества со стороны заверяющего лица в виде его отказа от подписи им документа. Открытый ключ подписи вычисляется, как значение некоторой функции от закрытого ключа, но знание открытого ключа не дает возможности определить закрытый ключ.

Главное свойство ключевой пары: по секретному ключу легко вычисляется открытый ключ, но по известному открытому ключу практически невозможно вычислить секретный.

Сообщения шифруются с помощью открытого ключа, а расшифровываются с использованием секретного ключа. Таким образом, расшифровать сообщение может только адресат (и отправитель) и больше никто. Использование асимметричных алгоритмов снимает проблему *распространения ключей* пользователей в системе, но ставит новые проблемы: *достоверность полученных ключей*.

На практике процесс генерации ключей в асимметричных системах начинается с выбора простых чисел. Для этого с помощью генератора случайных чисел находится произвольное нечётное число требуемой разрядности. Далее это число проверяется с помощью того или иного теста простоты. На сегодняшний день, как уже отмечалось выше, существует множество способов (алгоритмов) проверки является ли число простым или нет.

III. УСОВЕРШЕНСТВОВАННЫЕ АЛГОРИТМЫ ДЛЯ СИСТЕМЫ RSA

3.1. УСОВЕРШЕНСТВОВАННЫЙ ВЕРОЯТНОСТНЫЙ АЛГОРИТМЫ ОПРЕДЕЛЕНИЯ ПРОСТОТЫ ЧИСЛА

Поскольку по своей сути тест М-Р вероятностный, то в случае справедливости известной гипотезы Римана, есть возможность перевести этот тест в разряд детерминированных, если провести тестирование с использованием всех индикативных чисел из диапазона от 2 до $2(\log N)^2$. Основание какого логарифма фигурирует в данном случае - не известно, об этом авторы работ заставляют задуматься нас самих. Тому пример цитата из работы А.В. Черемушкина [1]: «**Теорема**. Если верна обобщённая гипотеза Римана и n является сильно псевдопростым по основанию a для всех чисел a из интервала $1 < a < 2(\log n)^2$, то n - простое число».

Сама собою напрашивается мысль, что указанный диапазон крайне завышен, в чем легко убедиться, если учесть только все простые числа из данного диапазона. Ведь, даже работая с числами 10^{1000}, мы не выходим за пределы уже известных простых чисел. Действительно, в приведённом случае нам придётся взять в качестве индикативных чисел все простые числа из ряда от 2 до, примерно, 20000000 (если предположить, что $\log N = \log_2 N$, то $2(\log_2 10^{1000})^2 \approx 2(1000 \cdot 3{,}5)^2$). Таблица простых чисел в этом диапазоне уже имеется и известна.

Но гипотеза Римана никем (официально) не доказана и неизвестно, когда это будет. Так что этот путь превращения вероятностного теста в детерминированный, можно сказать, тупиковый. Большую надежду на решение задачи мы связываем с наличием сильно псевдопростых чисел. По нашему мнению, здесь имеется хороший шанс превратить вероятностный тест в детерминиро-ванный в любом диапазоне чисел M. Это станет возможным, если:

a) будет найдено сильно псевдопростое число такого же порядка, что и число M,

b) будет найдена приемлемая аппроксимация зависимости $A_{кр}$ (точнее, количество итераций – $K_{кр}$) от длины сильно псевдопростого числа, доведённая до значений числа M.

Мы такую зависимость обнаружили, она имеет следующий вид:

$$K_{кр} = 0.5 \cdot \lg N. \qquad (2)$$

Выбор такого $K_{кр}$ хорошо согласуется с числовыми экспериментами в диапазоне проверяемых чисел, вплоть до 10^{57}. Как видно из (2), количество итераций здесь на несколько порядков меньше, чем вычисленное по выражению $2(\log N)^2$ в случае справедливости гипотезы Римана.

Следующим шагом, значительно сокращающим число операций при тестировании простоты числа, будет тактика выбора индикативных чисел, непосредственно примыкающих к $K_{кр}$.

IV. ДЕТЕРМИНИРОВАННЫЙ АЛГОРИТМ ОПРЕДЕЛЕНИЯ ПРОСТОТЫ ЧИСЛА

4.1. НОВЫЙ ПОДХОД К ПОСТРОЕНИЮ АЛГОРИТМА ОПРЕДЕЛЕНИЯ ПРОСТОТЫ ЧИСЛА НА ОСНОВЕ СОПОСТАВИТЕЛЬНОГО АНАЛИЗА

Существующие детерминированные алгоритмы тестирования чисел на простоту весьма сложны. Например, в работе Миллера [2] приведен алгоритм, который детерминированно проверяет простоту числа n за $0(n^{1/7})$ арифметических операций. Этот же алгоритм можно модифицировать так, что он будет делать $0(\log^4 n)$ арифметических операций; однако в этом случае его корректность опирается на справедливость расширенной гипотезы Римана, которая до сих пор так и не доказана! Сложность алгоритма Конягина-Померанса [3] оценивается тоже величиной $0(\log^4 n)$, но без выполнения упомянутой гипотезы Римана.

Предлагаемый авторами алгоритм простоты строится на основе исследования свойств диофантовых уравнений (по аналогии с подходами в алгоритмах Ферма), но с использованием методов сопоставительного анализа. Существенным моментом здесь являются новые, нетрадиционные для теории чисел, способы и подходы к решению классической задачи о разрешимости/неразрешимости исходного диофантового уравнения. По нашим предварительным оценкам сложность этого алгоритма по сравнению с известными детерминированными тестами не превосходит $0(\log^2 n)$.

4.2. МЕТОДИКА ИСПОЛЬЗОВАНИЯ ФОРМАНТНОГО АНАЛИЗА К РЕШЕНИЮ ДИОФАНТОВЫХ УРАВНЕНИЙ ПРИМЕНИТЕЛЬНО К КРИПТОАЛГОРИТМАМ RSA

Наш анализ исследования неразрешимости /разрешимости диофантового уравнения второго порядка ограничим рассмотрением задачи только о детерминированности теста простоты большóго числа. Решение поставленной задачи начнем с более общей - задачи о разложении большóго составного числа на два простых сомножителя.

Рассмотрим большóе составное число в виде произведения двух простых чисел $M=N_1*N_2$ (или, как в системе RSA: $n = q*p$). Далее будем исследовать, в качестве исходных, квадратичные уравнения нулевого уровня в следующих трёх видах (где M – факторизуемое число):

$$X*Y = M = N_1*N_2, \qquad (3)$$

$$X^2 = Y^2 + M, \qquad (4)$$

$$X^2 - 2X*Y = M. \qquad (5)$$

Во всех этих случаях нужно найти значения неизвестных X и Y. Задача облегчается уже тем, что изначально нам известен порядок чисел N_1 и N_2. Например, большая практика работы с ключами RSA показывает, что N_1 и N_2 (полагая $N_1 > N_2$) - числа одного порядка или отличаются на один, максимум на два порядка.

Поскольку в сопоставительном анализе мы имеем дело с формантами, то от алгебраических уравнений (1) - (3) перейдем к формантным следующих трёх видов (соответственно, к трем вычислительным схемам):

$$Fp(X)*Fp(Y) = Fp(M) \qquad (6)$$

$$Fp(X^2) = Fp(Y^2) + Fp(M) \qquad (7)$$

$$Fp(X^2) - 2*Fp(X*Y) = Fp(M) \qquad (8)$$

Какой схемой воспользоваться - определяет оператор. Решение задачи, очевидно, будет получено, после того, как мы найдём значения X и Y.

Так обстоят дела при решении обычного алгебраического уравнения. В сопоставительном же анализе, во многих случаях, достаточно найти только

форманты неизвестных по какому-либо основанию, что, как будет показано, намного проще.

Поставим задачу: разработать методику, и это - главное, нахождения собственных формант неизвестных. Действительно, предположим, например, для уравнения (4) и схемы (7), что уже найдены собственные форманты по некоторому большóму основанию p:

Тогда $X + Y = p(nX = \boldsymbol{p}n + X_0,\ Y = \boldsymbol{p}m + Y_0 + m) + (X_0 + Y_0) = N_1,\ X - Y = p(n - m) + (X_0 - Y_0) = N_2$.

Если $p > N_1$, то указанные форманты (суммы и разности) по основанию p являются *предельными*, поэтому находим сразу искомые числа:

$$N_1 = X_0 + Y_0,\quad N_2 = X_0 - Y_0.$$

Можно показать, что, в случае использования схемы (8) уравнения (5), необходимо обеспечить $p > M$, что, конечно же, намного сложнее, поэтому предпочтение следует отдавать первым двум схемам. Поясним процедуру нахождения собственных формант неизвестных на простом примере.

Пример 1. Пользуясь изложенным подходом, найдем разложение числа 143. Поскольку любое составное число можно представить, как

$$(X + Y)(X - Y),$$
$$X^2 - Y^2 = 143 = M. \qquad (9)$$

Легко убедиться, что X - чётное, а Y - нечётное. Действительно, $X^2 + 1 - Y^2 = 144$, из чего следует такое сопоставительное уравнение по основанию 2:

$$K_2(X^2 + 1),\ 2K_2(Y) = K_2(144) = 4$$

которое будет разрешимо, если X – число чётное, а Y - нечётное. Если же (9) переписать в виде:

$$X^2 = Y^2 - 1 + 144$$

и записать его сопоставительное уравнение по основанию 2

$$2K_2(X) = K_2(Y^2 - 1),\ 4 \geq 3, \qquad (10)$$

то оно будет разрешимо, если $K_2(X) \geq 2$, т.е. если делится только на 4 и больше.

Теперь найдем форманты неизвестных X и Y по основанию 5. Они таковы:

$$X^2 = 5k + (1,4) \text{ и } Y^2 = 5m + (1,4). \qquad (11)$$

Тогда $5k + (1,4) = 5m + (1,4) + (28 \times 5) + 3$

или

$$5k + (1,4) = 5(m + 28) + (1,4) + 3$$

или

$$5k + (1,4) = 5(m + 28) + (7, 4).$$

Из всех 4-х возможных вариантов, подлежащих исследованию, рассмотрим только вариант: $5k + (4) = 5(m+28) + (4)$, который и приводит к искомому решению. Покажем это.

Действительно, из этого равенства следует, что форманты будут тождественными, если $k = m + 28 = z$.

Далее, необходимо решать следующую систему из двух уравнений:

$$X^2 = 5z + (4), \qquad (12)$$

$$Y^2 = 5(z - 28) + (1). \qquad (13)$$

Решать эту систему будем путём «малого» перебора, учитывая, что X есть число чётное, а Y - нечётное, и z делится на 4 (см. (10) и (12)).

В сопоставительном анализе доказано, что для уравнений типа (9) справедливо двойное неравенство

$$1 + M < 2(X^2 + 2Y^2) \leq M(M + 1)$$

которое «поможет» нам в подборе решения. Имеем, подставив (12) и (13):

$$144 < 2(10z - 136) \leq 143 \cdot 144,$$
$$72 < (10z - 136) \leq 143 \cdot 72.$$

Но поскольку $72 + 136 \leq 10z \leq 143 * 72 + 136$, устанавливаем такое окончательное неравенство для z: $20 \leq z \leq 1044$. Но из форманты для Y^2 следует, что z должно быть не меньше 28.Таким образом, необходимо начать с $z = 28$ и далее перебирать все чётные z, кратные 4. Устанавливаем уже на первом шаге, что эта система разрешима при $z = 28$, т.е.

$X = 12$, $Y = 1$. $X - Y = 11$, $X + Y = 13$, $11 * 13 = 143$. Всё!

Если для формант неизвестных взять в качестве основания число, большее, чем 5, то можно сократить количество итераций.

Понятно, что процедура достаточно затратна, но при решении задачи факторизации действует своеобразный «закон» о том, что наиболее эффективные алгоритмы для больших чисел менее пригодны для работы с малыми числами.

Из вышесказанного ясно, что ключевым моментом для тестирования диофантового уравнения на **неразрешимость** является отыскание собственных формант неизвестных X и Y и по очень большим основаниям p. Поскольку величина p должна быть больше N_1 для схем (6) и (7) и больше M для схемы (8), то понятно, что напрямую решить такую задачу - найти собственную форманту X и Y по большому произвольному основанию p - невозможно, поскольку количество *нормативных операций* в этом случае сопоставимо с величиной числа M, в лучшем случае - N_2.

Например, предположим, что $N_2 \approx 10^{10}$. В этом случае полная форманта любого из неизвестных имеет скобку остатков длиной такого же порядка, что и число N_2 (то есть будет содержать, практически, столько же и ложных формант. Точнее - на единицу меньше). Поэтому, в общем случае, для избавления от всех ложных формант (процедура селекции) потребуется равное им количество нормативных операций, что, безусловно, неприемлемо! Но если рассматривать последовательно форманты по основаниям, равным сомножителям некоторой сложной (составной) форманты числа большой разрядности, и для каждой такой «малой» форманты находить собственную форманту, то количество нормативных операций потребуется на несколько порядков меньше!

Действительно, для выбора большого по величине составного основания, например, порядка $p \approx 10^{10}$, достаточно найти по отдельности всего 7 собственных формант по малым основаниям, например, 6,10,14,22,26,34,38. Произведение этих оснований равно 028.555.887.360 (11-разрядное число) и будем рассматривать это число, близкое к 10^{10}, как удовлетворяющее нашим требованиям к поиску большого основания p. Количество нормативных операций для нахождения собственных формант по малым основаниям в этом случае составит величину порядка суммы этих чисел, то есть:

$$6 + 10 + 14 + \ldots + 38 = 124,$$

что намного меньше количества нормативных операций для нахождения собственной форманты неизвестных X и Y по основания p порядка 10^{10}.

В реальности получить собственную форманту достаточно сложно. Возможно, что в некоторых случаях полной селекции не получится и останутся некоторые ложные форманты. Это значительно осложнит вычисление (по числу операций). Каждая ложная форманта, если их, к примеру, k, увеличивает число операций в k-раз.

Из вышесказанного следует, что узловым (существенным) моментом предлагаемой методики является процедура исключения ложных формант, базирующаяся на тестировании неразрешимости диофантовых уравнений методами сопоставительного анализа.

Заметим, что вопрос о нахождении эффективного алгоритма определения неразрешимости диофантового уравнения может являться темой отдельной научно-исследовательской работой.

V. УСОВЕРШЕНСТВОВАННЫЕ АЛГОРИТМЫ ГЕНЕРАЦИИ ПРОСТОГО ЧИСЛА

Цель улучшения алгоритма - создать такой ряд (набор, поле) целых чисел, в которых существенно увеличивается процент содержания простых чисел. Созданный ряд затем тестируется на простоту с помощью теста Миллера-Рабина.

Как известно (см. конец раздела 2.2), существующие алгоритмы поиска простого числа заданной разрядности выбирают случайное нечётное число *той же разрядности*, а затем прибавляют к этому числу (по возрастающей) все чётные числа (можно и наоборот). Каждый раз получившееся число проверяют тестом простоты (Миллера-Рабина).

Разрабатываемый авторами новый алгоритм ускоренной генерации простых чисел заданной разрядности позволяет свести их поиск к достаточно узкому диапазону от M до, примерно, $(8\ldots10)M$, т.е. в тестируемом диапазоне будут содержаться числа, отличающиеся, в крайнем случае, лишь на порядок, что соответствует требованиям RSA к свойствам ее криптоключей.

5.1. ОПИСАНИЕ АЛГОРИТМА

Задаётся некоторое число желаемой разрядности в виде суммы:

$$H = M + 2C!p, \qquad (*)$$

где

- $C!$ – т.н. субфакториал (т.е. произведение всех возможных комбинаций парных произведений простых чисел в интервале от 2 до некоторого $C_{MAX}/2$ или т.н. слабая половина факториала $C!$);
- p - варьируемый параметр, программно перебираемый в виде всевозможных произведений из чисел, входящих в субфакториал $C!$.
- M - произведение из двух простых чисел, из интервала $C_{MAX}/2$ до C_{MAX} (или т.н. сильная половина факториала $C!$).
- Выбор C_{MAX} зависит от разрядности генерируемых чисел.

В данном алгоритме используется известное свойство взаимно простых чисел: сумма двух взаимно простых чисел не будет содержать делителей этих чисел.

Рассмотрим, например, числовой ряд, полученный по формуле (*), где $M = 1001$, а $2C!p=30k$:

$$1001 + 30k, \text{ где } k = 1, 2, 3, 4, 5, 6, 8. \qquad (**)$$

Заметим, что $1001 = 7 \cdot 11 \cdot 13$.

В этом случае, на основе описанного алгоритма мы получаем ряд целых чисел, в которых плотность простых чисел достаточно велика (порядка 70%). Действительно, полученный ряд (**)

1031 1061 1091 1121 1151 1181 1241

содержит в себе 5 простых чисел из 7 (простые числа подчёркнуты).

В статье изложены материалы, составляющие лишь часть имеющихся у авторов теоретических и практических наработок в области использования инструментов и методов СпА для решения классических и современных задач теории чисел, представляющих интерес для программистов и разработчиков криптосистем.

ЗАКЛЮЧЕНИЕ

1. Используемый до настоящего времени традиционные методы для решения криптографических задач аппарат классической теории чисел, на наш взгляд, исчерпал себя. Будущее теоретической криптографии, точнее, её математической основы, мы связываем с сопоставительным анализом, с нахождением способов его применения к традиционным задачам закрытия информации.
2. Вместо усиленной эксплуатации Малой теоремы Ферма для поиска простого числа нами предлагается другое направление, а именно, использование уравнения Ферма $X^2 = Y^2 + N$.
3. Нами показано, что в случае неразрешимости этого уравнения, при условии, что $X - Y > 1$, число N есть ПРОСТОЕ.
4. Самая главная трудность состоит в том, что приходится иметь дело с большими основаниями формант неизвестных, что, в свою очередь, очень затрудняет нахождение собственных формант X и Y.
5. Генеральное направление, сулящее успех, нам видится в том, чтобы найти методы эффективной селекции ложных формант, что сводится, по существу, к нахождению способов доказательства НЕРАЗРЕШИМОСТИ конкретных диофантовых уравнений.
6. В статье изложены материалы, составляющие лишь часть имеющихся у авторов теоретических и практических наработок в области использования инструментов и методов СпА для решения классических и современных задач теории чисел, представляющих интерес для программистов и разработчиков криптосистем.

ЛИТЕРАТУРА

[1] А.В. Черёмушкин «Лекции по арифметическим алгоритмам в криптографии».- М:. МНЦНМО, 2002 г, - стр.51.

[2] Miller G.L. Riemann's hypothesis and tests for primality //J. Comput. and Syst. Sci. 1976. V. 13. P. 300—317. Перевод: Кибернетич. сборник. 1986. Вып. 23. С. 31-50 .

[3] Konyagin S.V., Pomerance C. On primes recognizable in deterministic polynomial time // Algorithms and combinatorics. Springer-Verlag, 1997. (The mathematics of Paul Erdos; V.). P. 176—198.