# ОЦЕНКА ЭФФЕКТИВНОСТИ АСИММЕТРИЧНЫХ КРИПТОСИСТЕМ, ОСНОВАННЫХ НА ТЕОРИИ ЧИСЕЛ

**Е. Ц. АЛАВЕРДЯН**

Применение асимметричных криптосистем упразднило проблему распределения криптографических ключей в процессе безопасного информационного обмена. В ситуациях, когда между сторонами обмена отсутствует доверие, точнее, когда секретный ключ не доверяется шифрующей стороне, асимметричные криптосистемы просто незаменимы.

Асимметричные криптосистемы позволяют легко устанавливать канал для обмена ключами между любыми двумя удаленными пользователями без личной встречи или привлечения службы аутентификации. В этом заключается важное преимущество асимметричных криптосистем перед стандартными методами. Действительно, что может быть привлекательнее для защиты информации, передаваемой по незащищенному каналу, чем способ шифрования, при котором все знают ключ шифрования и могут им воспользоваться для преобразования информации, в том числе и недобросовестные участники информационного обмена, но дешифровать ее может только единственный участник, который разработал алгоритм шифрования. Наряду с этим, для шифрования секретный ключ не используется - он необходим только при дешифровании. Отсюда и основная идея асимметричных криптосистем - предоставить каждой стороне информационного обмена свой собственный ключ.

Применение асимметричных криптосистем производится с учетом таких факторов, как защищенность, размер ключа и скорость вычисления. Бесспорно, наиболее важный показатель в выборе асимметричных криптографических технологий - это безопасность. В некотором смысле это выбор между разными верами, такими, как, например, вера в трудности факторизации

чисел, вычисление дискретных логарифмов по модулю простого числа или разложение составных алгебраических структур.

Асимметричные криптосистемы обеспечивают определенные услуги по защите информации. К их числу относятся передача конфиденциальной информации, обеспечивающей неразличимость сообщения; установление ключа, когда два лица, использующих открытый информационный канал, хотят согласовать секретный ключ для применения его в некоторой симметричной криптосистеме; идентификация системы, где пользователи доказывают, что они уполномочены иметь доступ к данным или к системным средствам, или что они - те, за кого себя выдают; невозможность отречения, обеспечивающая невозможность отрицания авторства сообщения, а также получения сообщения; аутентификация, позволяющая проверить целостность данных и источник сообщения; электронно-цифровая подпись.

Наиболее очевидным понятием безопасности для асимметричной криптосистемы принято считать то, что противник, имеющий открытый ключ и криптотекст, полученный применением того же открытого ключа, не способен в разумное время определить соответствующий оригинал. Однако, такое утверждение весьма слабо. Желательно предотвратить ситуацию, когда противник угадает любую информацию относительно оригинала криптотекста. Это понятие более сильное (так называемая “семантическая безопасность”) впервые выдвинуто Голдвассером и Мисали [1, 273].

Имеется несколько показателей криптостойкости, среди которых можно упомянуть:

- количество всех возможных ключей;
- среднее время, необходимое для криптоанализа.

Построение стойкой асимметричной криптосистемы предполагает существование специального класса математических функций, названных односторонними, впервые выдвинутых в работе Диффи и Хеллмана[2,110]. Функции, обладающие свой-

ством односторонности необходимы для того, чтобы обеспечить невозможность вычисления секретного ключа из открытого и придавать практическую стойкость криптографическим алгоритмам, путем сведения мощных атак к решению трудноразрешимых задач.

Стандартный способ доказательства высокой стойкости криптографических алгоритмов заключается в формальном доказательстве того факта, что атака на алгоритм эквивалентна решению одной из хорошо известных трудноразрешимых задач. Упрощенно идею односторонней функции можно представить на примере трафика движения по городским улицам с односторонним движением.

Одностороннюю функцию с секретом $f_t(x): D \to R$ легко вычислить для всех $x \in D$, но очень трудно обратить для почти всех значений из $R$. Однако, если используется секретная информация $t$, то для всех значений $y \in R$ легко вычислить величину $x \in D$, удовлетворяющую условию $y = f_t(x)$. Иначе говоря,

- вычисление $y = f_t(x)$. - задача полиномиального времени для любого пользователя, если $x$ и $t$ известны,
- вычисление $x = f_t^{-1}(y)$ - задача полиномиального времени для любого пользователя, если $y$ и $t$ известны,
- вычисление $x = f_t^{-1}(y)$ - задача неполиномиального време-ни для всех пользователей, если $y$ известно, но $t$ неизвестно.

Характерная особенность рассматриваемого процесса заключается в том, что сложность решения прямой задачи (вычисление $f_t(x)$) намного проще решения обратной задачи, которая может даже не иметь решения. Именно по этой причине, функция, описывающая рассмотренную выше ситуацию, называется односторонней.

Таким образом, для обладателей секрета решение обратной задачи имеет сложность, такую же, как и для прямой задачи, в то время как для всех остальных участников информационного

обмена, обратная задача может даже не иметь решения.

Односторонняя функция, на базе которой создан асимметричный криптографический алгоритм, должна обладать секретом $t$. Именно это обстоятельство позволяет обладателю секрета, в отличие от остальных, выполнить обратную операцию (дешифрование) за приемлемое время и с доступными для этого вычислительными ресурсами. Особенность такого процесса заключается в том, что открытый ключ не может использоваться для дешифрования и поэтому может быть передан по незащищенному каналу, с тем, чтобы им мог воспользоваться любой, желающий передать конфиденциальную информацию, в то время, как секретный ключ хранится только у владельца сгенерированных ключей.

Наиболее важная односторонняя функция, используемая в асимметричной криптографии - это разложение на множители или факторизация целых чисел на том основании, что определение множителей является очень трудоемкой вычислительной операцией.

Другой важный класс трудноразрешимых задач связан с дискретным логарифмированием. Предположим, что $(G,\bullet)$ представляет собой конечную абелеву группу, например, мультипликативная группа конечного поля или эллиптическая кривая над конечным полем. Проблема вычисления дискретных логарифмов состоит в определении целого числа $x$, которое при данных $A,B \in G$ удовлетворяет соотношению $A^x = B$.

***Криптосистема RSA.*** Наиболее известной асимметричной криптосистемой является криптосистема RSA, названный по первым буквам фамилий своих создателей. Криптосистема RSA[3,124] – первая практическая реализация криптографии на основе понятия односторонней функции с секретом, предложенного Диффи и Хеллманом [2,109]. Криптосистема RSA описывается алгоритмом, приведенным ниже.

***Генерация криптографических ключей.*** Для создания пары криптографических ключей, пользователю необходимо выпол-

нить следующие операции:

1. Выбрать два случайных простых числа $p$ и $q$, удовлетворяющих условию $|p| \approx |q|$.
2. Вычислить $N = pq$.
3. Вычислить $\Phi(N) = (p-1)(q-1)$
4. Выбрать случайное целое число $e < \Phi(N)$, удовлетворяющее условию $gcd(e, \Phi(N)) = 1$, и найти целое число $d$, такое что

$$ed \equiv 1 (mod(\Phi(N)).$$

Поскольку $gcd(e, \Phi(N)) = 1$, это уравнение имеет решение $d$, которое можно найти с помощью расширенного алгоритма Евклида [4,205].

5. Используя пару $(N, e)$ в качестве параметров открытого ключа, тщательно уничтожить числа $p, q, \Phi(N)$ и запомнить число $d$ в качестве закрытого ключа.

***Шифрование информации.*** Для того, чтобы переслать секретное сообщение, имеющего длину $m < N$, создается зашифрованный текст $c$.

$$c \equiv m^e \ (mod N).$$

***Дешифрование информации.*** Для того чтобы дешифровать зашифрованный текст $c$, вычисляется следующее:

$$m \equiv c^d \ (mod N).$$

Из определения модулярной арифметики [4,206] следует, что сравнение $ed \equiv 1 (mod(\Phi(N))$ в этом алгоритме эквивалентно уравнению

$$ed = 1 + k\Phi(N),$$

где $k$ - некоторое целое число.

Следовательно, значение, вычисляемое первой стороной в результате выполнения процедуры дешифрования, определяется по следующей формуле:

$$c^d \equiv m^{ed} \equiv m^{1+k\Phi(N)} \equiv m \cdot m^{k\Phi(N)} (mod N), \qquad (1)$$

что указывает на криптографическую природу алгоритма.

Отметим, что неравенство $m < N$ практически всегда означает, что $m \in Z_N^*$. То есть, почти все числа, которые меньше числа $N$, принадлежат мультипликативной группе целых чисел, взаимно простых с числом $N$. Условие $m \in Z_N^*$ нарушается, если $m = up$ или $m = vq$, где $u < q$ и $v < p$. В этих ситуациях можно разложить число $N$ на простые множители, вычислив значение $gcd(m, N)$. Предполагая, что это является трудноразрешимой задачей, можно предположить, что любое сообщение $m < N$, созданное получателем, удовлетворяет условию $m \in Z_N^*$.

Если $m \in Z_N^*$, то по теореме Лагранжа [4,204]

$$ord_N(m) \mid Z_N^* = \Phi(N),$$

где $ord_N(m)$ – это порядок числа $m(mod N)$.

Это утверждение справедливо для всех $m \in Z_N^*$. В соответствии с определением порядка группы [4,203], это означает, что для всех $m \in Z_N^*$ выполняется условие

$$m^{\Phi(N)} = 1(mod N).$$

Отсюда следует, что $m^{k\Phi(N)} = (m^{\Phi(N)})^k = 1(mod N)$ для любого целого числа $k$. Итак, величина в формуле (1) действительно равна числу $m$.

Следует отметить, что операция деления по модулю не сводится к обычному делению [5,556], поскольку $0 \le a, b \le n$, а значит,, $-n < a \pm b < 2n$. Следовательно,

$$a \pm b(mod n) = \begin{cases} a \pm b, & \text{если } 0 \le a \pm b < n, \\ a \pm b - n, & \text{если } a \pm b \ge n, \\ n + (a \pm b), & \text{если } a \pm b < 0. \end{cases}$$

Оценка сложности основных операций в модулярной арифметике приводятся на Таблице 1.1, приведенной ниже.

Таблица 1.1. Поразрядные оценки сложности основных операций в модулярной арифметике.

| Операция над $a, b \in_U [1, n)$ | Сложность |
|---|---|
| $a \pm b (mod n)$ | $O_B(\log n)$ |
| $a \cdot b (mod n)$ | $O_B((\log n)^2)$ |
| $b^{-1} (mod n)$ | $O_B((\log n)^2)$ |
| $a/b (mod n)$ | $O_B((\log n)^2)$ |
| $a^b (mod n)$ | $O_B((\log n)^3)$ |

На практике, криптосистема RSA типично используется для шифрования таких коротких сообщений, как, например, номера кредитных карточек или для шифрования случайного ключа $k$, который в свою очередь, используется в симметричной схеме шифрования типа AES с целью шифрования сообщения. Ключ $k$ обычно весьма короток: например, $128$, $192$ или $256$ бита для AES, и поэтому может быть рассмотрен как целое число $M$ в интервале $[0, N-1]$[6,8].

Для шифрования таких единиц сообщения $M$ отправитель вычисляет криптотекст $C$, который является наименьшим положительным остатком $M^e$ по модулю $N$. Для дешифрования $C$, получатель вычисляет наименьший положительный остаток $C^d$ по модулю $N$[7,14].

Средства защиты криптосистемы $RSA$ от атак основаны на сложности вычисления корня *e*-ой степени шифрованного текста $c$ по составному целочисленному модулю $n$. Это - так называемая задача RSA.

***Криптосистема Рабина.*** Криптосистема, разработанная Рабином, основана на сложности вычисления квадратного корня по модулю составного числа [8,23]. Работа Рабина носила теоретический характер. В ней впервые приводилось доказательство стойкости асимметричных криптосистем. Стойкость криптосистемы Рабина эквивалентна сложности задачи разложения целых

чисел на множители. Следует отметить также, что эквивалентность разрешимости задачи RSA и разрешимости задачи разложения целых чисел на множители еще не доказана. Алгоритм шифрования в криптосистеме Рабина эффективен и пригоден для многих практических приложений, например, для шифрования с помощью портативных устройств.

Криптосистема Рабина описывается следующим алгоритмом:

***Генерация криптографических ключей.*** Для создания пары криптографических ключей сторона А должна выполнить следующие действия:

1. Выбрать два случайных простых числа $p$ и $q$, удовлетворяющих условию $|p| \approx |q|$.
2. Найти число $N = pq$.
3. Извлечь случайное целое число $b \in_U \mathbb{Z}_N^*$.
4. Использовать пару $(N, b)$ в качестве параметров открытого ключа и запомнить пару $(p, q)$ в качестве параметров секретного ключа.

**Шифрование информации.** Для того чтобы послать стороне А секретное сообщение $m \in \mathbb{Z}_N^*$, сторона Б создает зашифрованный текст $c$:

$$c \leftarrow m(m + b)(mod\ N).$$

**Дешифрование информации.** Для того чтобы дешифровать зашифрованный текст $c$, сторона А решает квадратное уравне-ние

$$m^2 + bm - c \equiv 0(mod\ N), \qquad (2)$$

где $m < N$.

Из элементарной математики известно, что общее решение квадратного уравнения имеет вид:

$$m \equiv \frac{-b \pm \sqrt{\Delta_c}}{2}(mod\ N), \qquad (3)$$

где $\Delta_c \stackrel{def}{\equiv} b^2 + 4c(mod\ N)$.

Поскольку число $c$ зависит от элемента $m \in \mathbb{Z}_N^*$, квадратное уравнение $m^2 + bm - c \equiv 0(mod\ N)$, имеет решение в группе

$\mathbb{Z}_N^*$. Одним из этих решений является число $m$, посланное стороной Б. Отсюда следует, что число $\Delta_c$ должно быть квадратичным вычетом по модулю $N$, т.е. элементом группы $QR_N$, которое представлает собой множество квадратичных вычетов по модулю $N$.

Вычисления, связанные с дешифрованием, содержат операцию извлечения квадратного корня по модулю $N$. Вычислительная сложность этой задачи эквивалентна факторизации числа $N$. Следовательно, сторона А является единственным пользователем, который может вычислить формулу (3), поскольку только ей известны простые множители - компоненты числа $N$ на простые множители, и найти число $\sqrt{\Delta_c}$ используя алгоритм вычисления квадратных корней по составному модулю.

Для каждого зашифрованного текста $C$, посланного стороной Б, существует четыре разных значения $\sqrt{\Delta_c}$ и, следовательно, существуют четыре разных результата шифрования. Как правило, реальное исходное сообщение содержит избыточную информацию, позволяющую стороне А отличать правильные тексты от неправильных. В алгоритме шифрования Рабина используется только одно умножение и одно сложение. Следовательно, этот алгоритм работает быстрее, чем алгоритм шифрования RSA. Отметим, что стойкость криптосистемы Рабина, как и в случае RSA, зависит от сложности разложения целых чисел на простые множители [9,112].

***Криптосистема Эль-Гамаля.*** Эль-Гамаль разработал весьма остроумную криптосистему с открытым ключом [10,470], использующую одностороннюю функцию с секретом Диффи-Хеллмана. Работа Эль-Гамаля вызвала большой теоретический и практический интерес, который сохраняется до сих пор.

Широкая популярность этой криптосистемы объясняется использованием задачи Диффи-Хеллмана, неразрешимость которой общепризнана. Считается, что ее сложность эквивалентна сложности задачи вычисления дискретного логарифма, которая, в свою очередь, является альтернативой задачи разложения це-

лого числа на простые множители, лежащей в основе криптосистем RSA и Рабина.

Криптосистема Эль-Гамаля действительно является криптографической, т.е. в результате дешифрования восстанавливается тот самый исходный текст, который был послан отправителем.

Поскольку

$$c_1^x \equiv (g^k)^x \equiv (g^x)^k \equiv y^k \equiv \frac{c_2}{m} (mod\ p),$$

формула

$$m \leftarrow \frac{c_2}{c_1^x} (mod\ p) \qquad (4)$$

позволяет восстановить исходный текст $m$.

Для деления, выполняемого в формуле (4), необходимо применить расширенный алгоритм Евклида, что в общем случае сопровождается большими вычислительными затратами, чем умножение. Однако отправитель может избежать деления, использовав следующие вычисления:

$$m \leftarrow c_2 c_1^{-x} (mod\ p)\ .$$

***Генерация криптографических ключей.*** Для создания пары криптографических ключей необходимо выполнить следующие действия:

1. Выбрать случайное простое число $p$.
2. Вычислить случайный мультипликативный порождаю-щий элемент $g \in \mathbb{F}_p^*$.
3. Извлечь случайное целое число $x \in_U \mathbb{Z}_{p-1}^*$ и считать его своим секретным ключом.
4. Вычислить открытый ключ, $y \leftarrow g^x (mod\ p)$.
5. Использовать тройку $(p, g, y)$ в качестве параметров открытого ключа и запомнить число $x$ в качестве секрет-ного ключа.

***Шифрование информации.*** Для того чтобы послать стороне А секретное сообщение $m < p$, сторона Б извлекает случайное

целое число $k \in_U \mathbb{Z}_{p-1}^*$ и вычисляет зашифрованный текст $(c_1, c_2)$:

$$\begin{cases} c_1 \leftarrow g^k (mod\ p), \\ c_2 \leftarrow y^k m (mod\ p). \end{cases} \quad (5)$$

***Дешифрование информации.*** Для того, чтобы дешифровать зашифрованный текст $(c_1, c_2)$, сторона А вычисляет формулу

$$m \leftarrow \frac{c_2}{c_1^x} (mod\ p).$$

Алгоритм шифрования (5) является вероятностным: он использует случайное число $k \in UZ_{p-1}$ , где $U$ указывает на равномерное распределение значений числа $k$. Допустим, что закрытый ключ $x$ стороны А является числом, взаимно простым с числом $p-1$ . Тогда закрытый ключ стороны А представляет собой число $y \equiv g^x (mod p)$, как и число $g$, является порождающим элементом группы $F_p^*$. Следовательно, число $y^k (mod p)$ пробегает всю группу $F_p^*$, когда элемент $k$ пробегает группу $Z_{p-1}$. Поскольку умножение по модулю $p$ являетсв перестановкой над группой $F_p^*$, для любого исходного сообщения $m \in F_p^*$ число $c_2 \equiv y^k m (mod p)$ пробегает группу $Z_{p-1}$. Итак, если $k \in UZ_{p-1}$, то $c_2 \in UF_p^*$. Это означает, что шифрование Эль-Гамаля *равномерно* распределяет исходное сообщение по всему пространству сообщений. А это, в свою очередь означает, что криптосистема является стойкой тогда и только тогда, когда задача Диффи – Хеллмана является трудноразрешимой. Как и в протоколе обмена ключами Диффи – Хеллмана, криптосистема Эль – Гамаля функционирует в подгруппе группы $F_q$ (конечное поле $q$ элементов), порядок которой является большим простым числом, или в большой группе точек эллиптической кривой,

определенной над конечным полем [11,590].

***Недостатки асимметричных криптосистем основанных на теории чисел.*** Из вышеизложенного следует, что стойкость существующих теоретико-числовых асимметричных алгоритмов приравнивается к сложности факторизации целых чисел и сложности вычисления дискретных логарифмов на конечном поле.

Детальный анализ вышеупомянутых алгоритмов производится и в настоящее время, особенно касательно их практической эффективности. Под практической эффективностью криптосистем мы подразумеваем рассмотрение проблемы с точки зрения скорее реальности их применения, чем асимптотической сложности применяемых вычислений. Эти алгоритмы являются наилучшими до настоящего времени: они общепризнанны и представляют собой большой криптографический интерес.

Однако, наряду с оценками стойкости существующих асимметричных криптосистем на основе теории чисел, другие аспекты этих криптосистем также рассматриваются, в частности, эффективность их построения.

Результаты анализа алгоритмов факторизации целых чи-сел и вычисления дискретного логарифма, приведенных в [6] и [7], указывают на то, что для достижения уровня стойкости, рав-ной факторизации целых чисел с 100 и 155 цифр (332 и 512 битов), стойкость дискретного логарифма в $GF(2^m)$ опять же требует битовую строку длиной примерно в 400 и 700 соответственно. Оценка алгоритмических парамтеров, размер результирующей линейной системы, а также, количество операций - протабулированы для этих вычислений, поэтому и уделяется большее внимание на относительную сложность вычисления вышеупомянутых задач в противоположность их абсолютному времени прогона.

В начале статьи были перечислены определенные услуги по защите информации, предоставляемые асимметричными криптосистемами. Касательно услуг, представленных пунктами 2 - 5, все перечисленные асимметричные криптосистемы применяются успешно и, в

этом отношении, они конкурентноспособны.

Тем не менее, элегантно решив проблему согласования и распределения криптографических ключей, как правило, криптографические функции в асимметричных криптосистемах действуют в очень крупных алгебраических структурах, т.е. они связаны с выполнением весьма затратных алгебраических операций. Применяемые на практике алгоритмы используют трудоемкие математические расчеты над большими числами. При этом, используемые сложные математические преобразования чисел, имеющих сто и больше цифр, требуют огромных затрат машинного времени и ресурсов и приводят к снижению быстродействия криптосистемы. В приложениях, особенно связанных с передачей конфиденциальной информации больших объемов, теоретико-числовые асимметричные криптосистемы не обеспечивают требуемой производительности и неизбежно включают быстродействующие симметричные алгоритмы. В результате, существующие асимметричные алгоритмы применяются только для выполнения вспомогательных (относительно процесса обеспечения секретности) функций, таких, как цифровая подпись и шифрование ключей, применяемых симметричными алгоритмами.

Надежды на то, что с ростом быстродействия технических средств разрыв между быстродействием асимметричных и симметричных алгоритмов будет сокращаться, безосновательны. Это происходит потому, что с ростом быстродействия технических средств, при сохранении размера чисел, снижается степень защищенности алгоритма, а это, в свою очередь, приводит к необходимости увеличить размерность вычислений. Поэтому, теоретико-числовые алгоритмы, как направление в криптографии, которое привело к появлению большого числа оригинальных асимметричных решений, создало также иллюзию, что термины «асимметричный алгоритм» и «трудоемкие вычисления» - синонимы.

Для оценки вычислительной сложности теоретико-числовых алгоритмов рассмотрим два аспекта:

- процесс шифрования/дешифрования,

- генерация криптографических ключей.

Оба процесса, шифрование и дешифрование в асимметричных криптосистемах включают возведение целого числа в степень другого целого числа с вычислением $mod n$. Если экспоненциация произведена на целые числа, а потом приведена $modulo n$, промежуточные вычисления могут быть гигант-скими.

Для вычисления значения $a^m$, где $a$ и $m$ оба положительные целые числа, $m$ выражается через двоичное число $b_k, b_{k-1}, \ldots, b_0$, так что

$$m = \sum_{i=0}^{k} b_i\, 2^i \ .$$

Из этого следует, что

$$a^m = a^{\sum_{i=0}^{k} b_i 2^i} = \prod_{i=0}^{k} a^{b_i 2^i} = \prod_{\substack{i=0 \\ b_i \neq 0}}^{k} a^{(2^i)},$$

$$a^m mod n = \left[\prod_{b_i \neq 0} a^{(2^i)}\right] mod n = \left(\prod_{b_i \neq 0} \left[a^{(2^i)} mod n\right]\right) \mathrm{mod} n.$$

Очевидно, что экспоненциация достаточно трудоемкая процедура, особенно при больших значениях $m$.

***Генерация пар криптографических ключей.*** Перед применением асимметричной криптосистемы каждая сторона информационного обмена генерирует пару криптографических ключей. Процесс включает следующие задания:

- Сгенерировать два простых числа, $p$ и $q$.
- Выбрать первый положительный компонент пары $(e, d)$ и вычислить второй компонент.

Рассмотрим сперва выбор $p$ и $q$. Так как значение $n = pq$ будет известен всем потенциальным оппонентам, то для предотвращения раскрытия $p$ и $q$ методом перебора, эти простые числа должны быть выбраны из множества достаточно больших положительных целых чисел. С другой стороны, метод, используе-

мый для генерации простых чисел, должен быть разумно эффективным.

До настоящего времени не существует эффективного математического метода генерации простых чисел произвольной величины. Разработка новых решений в этой области насущный вопрос. Процедура, которая, как правило, практикуется, представляет собой генерацию нечетного числа величиной желаемого порядка, далее производится тестировние этого же числа на простоту. Если результат тестирования на простоту оказывается неудачным, то выбирается следующее по порядку нечетное число и процедура повторяется снова, пока требуемое число не определится.

До настоящего времени разработано целое множество тестов на простоту, такие, как например, алгоритмы [KNUTH 98], но все эти тесты без исключения - вероятностные. Это значит, что эти тесты просто определяют, что данное число *вероятно* простое. Несмотря на недостаток определенности или уверенности в простоте, эти тесты могут прогнаться таким образом, чтобы все же приблизить эту вероятность к желанному $1.0$. Примером этому может служить один из наиболее эффективных и популярных алгоритмов, называемый методом Миллера-Рабина. Для тестирования числа на простоту выполняется некоторое вычисление, которое включает число $n$ и произвольно выбранное положительное цело число $a$. Если число $n$ делится на число $a$, то есть тестирование $n$ претерпевает неудачу, то число $n$ - не простое. Но, а если число $n$ тестируется удачно, то на этом этапе тестирования это лишь означает, *что $n$ либо простое, либо – не простое*. В таком случае это же число $n$ проходит многие другие аналогичные тесты с произвольно выбранными числами $a$, и только после всех, без исключения, положительных результатов тестирования констатируется, что данное число $n$, вероятно, простое.

Обобщая процедуру генерирования простого числа, имеем следующий вероятностный алгоритм:

1. Генерируется произвольное нечетное число $n$ требуемого порядка с использованием генератора псевдопростых чисел.

2. Выбирается произвольное $a < n$.

3. Производится вероятностный тест на простоту. Если процесс претерпевает неудачу, выбранное значение $n$ отбрасывается, а процедура генерации возобновляется с шага 1.

4. Если число $n$ многократно протестировано удачно, его значение *фиксируется как простое*, в противном случае процедура возобновляется с шага 2.

Это, конечно, трудоемкая процедура. Ситуация спасена, если этот процесс выполняется относительно нечасто: лишь тогда, когда необходимо сгенерировать новую пару криптографиических ключей. В стратегических системах быстрого реагирования, таких, как например, динамически переконфигурируемые структуры, действующие при чрезвычайных ситуациях, низкая производительность генерации пар криптографических ключей, а также процессов шифрования/дешифрования, применение теоретико-числовых асимметричных криптографических систем отодвигается во избежание недопустимой потери времени принятия оперативных решений.

Стоит еще задаться вопросом: сколько чисел могут быть отброшены, до того как подходящее простое число будет найдено? Известная теорема простых чисел утверждает, что простые числа, близкие к $N$, распределены в среднем на каждую $(\ln N)$ целых чисел [3]. Это означает, что, в среднем, необходимо протестировать порядка $(\ln N)$ целых чисел до того, как найти одно простое число. Отбросив все четные числа, этот порядок выравнивается к $(\ln N)/2$. Например, при необходимости генерации простого числа величиной $2^{200}$, приблизительно $\frac{\ln(2^{200})}{2} = 70$ проб может понадобиться для нахождения простого числа.

Определив простые числа $p$ и $q$, процесс генерации криптографических ключей завершается выбором значения $e$ и вичис-

лением $d$, или же наоборот, выбором значения $d$ и вичислением $e$. Предполагая первое, необходимо выбрать $e$ таким образом, чтобы $\gcd(\Phi(n), e) = 1$, после чего вычислить $d \equiv e^{-1} \bmod \Phi(n)$. К счастью, существует один единственный алго-ритм, который одновременно вычисляет наибольший общий делитель двух целых чисел, и, если этот делитель равен еди-нице, то определяется обратное одного из двух целых чисел по модулю другого. Этот алгоритм широко известен под названием расширенного Евклидового алгоритма. Таким образом, процедура генерации серии произвольных чисел, их итерактивное тестирование по отношению к $\Phi(n)$ до успешного нахождения числа взаимно простое с $\Phi(n)$, успешно практикуется.

Теперь задаемся следующим вопросом: сколько произвольных целых чисел должно быть протестировано, для того, чтобы найти подходящее число, то есть число, взаимно простое с $\Phi(n)$? Из теории чисел известно, что вероятность того, что два произвольно выбранные целые числа взаимно простые, равно приблизительно 0.6 [3,125]. То есть, опять же немалое количество тестов необходимо произвести для нахождения подходящего целого числа, взаимно простого с $\Phi(n)$.

Криптографический алгоритм RSA использует только один тип вычислений - возведение в степень. Показатель степени определяет длительность выполнения процедуры вычислений. Чтобы обеспечить требуемый уровень стойкости, показаель степени, являющийся секретным ключом, должен быть достаточно большим, поэтому для вычисления требуется значительное машинное время.

Итак, проведен сравнительный анализ на стойкость и быстродействие между различными теоретико-числовыми асимметричными криптосистемами. Показано, что асимметричные криптосистемы, основанные на теории чисел, обладают достаточно высоким уровнем стойкости к существующим видам атак. Стойкость обеспечивается за счет применения ими теории делимости, факторизации целых чисел и дискретного логариф-

мирования, известными как классы труднорешаемых задач. Показано, что существенный недостаток асимметричных криптосистем, основанных на теории чисел – это низкий уровень их быстродействия по причине трудоемких вычислений, требующих огромных временных и вычислительных ресурсов и тем самым ограничивающих применение таких криптосистем только для генерации цифровой подписи и распределения криптографических ключей. Выявлено, что для передачи конфиденциальной информации особо больших объемов теоретико-числовые алгоритмы не пригодны из – за их низкой производительности.

**Ключевые понятия:** *асимметричная криптосистема, односторонняя функция, открытый и секретный ключ, эффективность криптосистемы*.

## ЛИТЕРАТУРА

**1. S. Goldwasser and S. Micali.** "Probabilistic encryption", Journal of Computer and System Sciences, 29 (1984), pp. 270-299.

**2. W. Diffie and Hellman.** "Multiuser cryptografic technics", In Proceedings of AFIPS 1976 NCC,. AFIPS Press, Montvale, N.J., 1976, pp. 109-112.

**3. R.L. Rivest, A. Shamir, and L. Adleman. "**A method for obtaining digital signatures and public key cryptosystems", Communications of the ACM, 21(2), 1978, pp. 120-126**.**

**4. Lang S.** "Algebraic Number Theory", Springer-Verlag. New-York, 1986, pp. 200-207.

**5. M. Wiener**. "Cryptanalysis of short RSA secret exponents", IEEE Transactions on Information Theory, 1990. pp 553-558.

**6. D. Boneh and G. Durfee. "**Cryptanalysis of RSA with private key *d* less than $n^{0.292}$ ", In J. Stern, editor, Advances in Cryptology – Proceedings of EUROCRYPT'99, Lecture Notes in Computer Science 1592, Springer-Verlag, 1999, pp 1-11.

**7. S. Cavallar,** B. Dodson, A.K.Lenstra, W. Lioen and Zimmer-mann. "Factorization of a 512 – bit RSA modulus", Advances in Cryptology – Proceedings of EUROCRYPT'00, Lecture Notes in Computer Science 1807, Springer – Verlag, 2000, pp 1-18.
**8. M.O. Rabin**. "Digitized signatures and public key functions as intractable as factorization", Technical Reports LCS/TR-212, MIT Laboratory for Computer Science, 1979, pp. 1-25.
**9. C. Pomerance**. "Analysis and comparison of some integer factoring algorithms", in Computational Methods in Number Theory, H.W. Lenstra, Jr. and R. Tijdeman(eds.), Math.Centrum Tract 154, 1982, 89-139.
**10.T. ElGamal.** "A public key cryptosystem and a signature scheme based on discrete logarithms", IEEE Transactions on Information Theory, IT-31(4), July 1985, pp. 469-472.
**11. D. Coppersmith**. "Fast evaluation of logarithms in fields of characteristic two", IEEE Transactions on Information Theory IT-30(4), July 1984, pp. 587-594.

**Ե. Ծ. ԱԼԱՎԵՐԴՅԱՆ**

## ԹՎԵՐԻ ՏԵՍՈՒԹՅԱՆ ՎՐԱ ՀԻՄՆՎԱԾ ԱՆՀԱՄԱՉԱՓ ԳԱՂՏՆԱՀԱՄԱԿԱՐԳԵՐԻ ԱՐԴՅՈՒՆԱՎԵՏՈՒԹՅԱՆ ԳՆԱՀԱՏՈՒՄԸ

*Ամփոփում*

Ներկայացված են թվերի տեսության վրա հիմնված անհամաչափ գաղտնահամակարգերի գործողության սկզբունքները, բերված են գաղտնագրման և վերծանման ընթացակարգերի նկարագրությունը, գաղտնագրային բանալիների զույգերի գեներացման մանրամասները՝ հետազոտվող գաղտնահամակարգերից յուրաքանչյուրի համար: Նշված գաղտնահամակարգերի արդյունավետության քանակական գնահատումը հնարավորություն է տալիս եզրակացնելու, որ թվերի տեսության վրա հիմնված գաղտնահամակարգերն, առանց բացառության, ցածր արդյունավետություն ունեն և կիրառելի չեն առանձնակի մեծ

չափերի տեղեկույթի գաղտնագրման ու վերծանման ընթացակարգերի համար:

**Առանցքային հասկացություններ.** *անհամաչափ գաղտնահամակարգ, միակողմանի ֆունկցիա, բաց և գաղտնի բանալիներ, գաղտնահամակարգի արդյունավետություն:*

**Y. Ts. ALAVERDYAN**

## ESTIMATION OF EFFICIENCY OF ASYMMETRIC CRYPTOSYSTEMS BASED ON THE THEORY OF NUMBERS

*Summary*

The principles of performance of asymmetric cryptosystems are given based on the theory of Numbers. For each of the explored cryptosystem the procedures of encryption and decryption, also of the cryptographic key pairs' generation details are presented. A quantitative analysis of the efficiency of the given asymmetric cryptosystems makes it possible to conclude that all the asymmetric cryptosystems based on the Number theory possess a low level of performance and cannot be applied for encryption and decryption of huge volume of information, which restricts their area of application.

**Key concepts:** *public key cryptosystem, one-way function, public key, private key, efficiency of the cryptosystem.*