HOW NUMBER THEORY HELPS CRYPTOGRAPHY

V. A. USPENSKY

Is it possible to invent such a method of encoding that everyone could cipher a message, and nobody but those who know the secret could decode the message? This paper offers a solution to that problem which is given by number theory.

Можно ли придумать такой способ шифрования, чтобы шифровать сообщения мог каждый, а расшифровать – только знающий секретный ключ? В статье рассказывается о решении, которое теория чисел предлагает для поставленной задачи.

КАК ТЕОРИЯ ЧИСЕЛ ПОМОГАЕТ В ШИФРОВАЛЬНОМ ДЕЛЕ

В. А. УСПЕНСКИЙ

Московский государственный университет им. М.В. Ломоносова

1. ПОСТАНОВКА ЗАДАЧИ

Когда участники процесса обмена сообщениями не желают, чтобы их сообщения были поняты кем-либо посторонним, они прибегают к шифрам. Сосредоточимся на тех важных случаях, когда среди участников выделяются один Получатель и один или несколько Отправителей. Потребуем выполнения следующих условий.

1. Расшифровать зашифрованное сообщение может только Получатель. Ключ к расшифровке настолько секретен, что неизвестен даже Отправителям.

2. Напротив, ключ к шифровке публикуется Получателем совершенно открыто, так что зашифровать сообщение может всякий желающий. Такой ключ называется *открытым ключом*.

Всякая система шифровки и дешифровки, удовлетворяющая сформулированным выше двум требованиям, называется *системой тайнописи с открытым ключом* (по-английски: public-key cryptosystem). Естественно встает вопрос, возможна ли подобная система. Оказывается, да, возможна [5, n° 6.2]. Одна их таких систем была указана тремя математиками, Ривéстом (R.L. Rivest), Шамиром (F. Shamir) и Адлеманом (L.M. Adleman), в их совместной публикации [4]. По начальным буквам фамилий система носит название *Система РША* (в оригинале RSA). Она опирается на факты из теории чисел. Цель настоящей статьи – изложить в доступной форме Систему РША.

2. ОБЩИЕ ПРЕДСТАВЛЕНИЯ О ШИФРАХ

2.1. Сообщения и шифрограммы

“Шифрование производится путем замены целых фраз, слов, слогов или отд. букв цифрами или буквами в различных комбинациях на основе заранее принятой системы, являющейся соответственно ключом для расшифровки текста”, – указывает Большая Советская Энциклопедия (изд. 3-е, т. 29, стлб. 1241). Эта формулировка не совсем точна: ведь при тайнописи с открытым ключом система шифрования не является ключом для расшифровки.

В общем случае ситуация выглядит так. Сообщение шифруется Отправителем и расшифровывается Получателем. Таким образом, от Отправителя к Получателю идет не само исходное сообщение M, а его зашифрованный образ C, который будем называть

шифрограммой исходного сообщения. Получив шифрограмму C, Получатель восстанавливает M. Перескажем то же на математическом языке. В распоряжении Отправителя имеется *шифровальный ключ*, то есть функция E, преобразующая сообщение M в шифрограмму C. В распоряжении Получателя имеется *дешифровальный ключ*, то есть функция D, преобразующая шифрограмму C в исходное сообщение M, так что $D(E(M)) = M$, откуда $E(D(C)) = C$.

Каждое сообщение есть конечная последовательность каких-то символов: букв, цифр, знаков препинания и т.п. (Пробел есть особый знак препинания; удобно изображать его каким-либо символом, например, решеткой #; тогда, скажем, фраза “4 мая” запишется так: “4#мая”.) Все эти символы выбираются из заранее указанного конечного списка. Любой такой список символов называется *алфавитом*, члены этого списка – *буквами*, а произвольная конечная цепочка букв – *словом в данном алфавите*. (Так, выражение “ъъыЩйь” есть слово в русском алфавите.) Если расширить русский алфавит за счет цифр, знаков препинания и некоторых других стандартных знаков, то любой русский текст можно считать словом в этом расширенном алфавите. А любой английский, французский и т.д. текст можно считать словом в расширенном латинском алфавите. Теперь ясно, что каждое сообщение есть слово в подходящем *Алфавите Сообщений*. А каждая шифрограмма есть слово в *Алфавите Шифрограмм*.

2.2. Переход к двоичному алфавиту

Вот простое и замечательное соображение: можно ограничиться двоичным алфавитом, то есть алфавитом, состоящим всего лишь из двух букв 0 и 1; слова в таком алфавите называются *двоичными*. В самом деле, пусть в рассматриваемом нами алфавите m букв и пусть $2^s \geq m$. Количество слов длины s в двоичном алфавите {0, 1} равно 2^s; поскольку этих слов оказалось не меньше, чем букв в исходном алфавите, то каждую букву исходного алфавита можно взаимно однозначно закодировать в виде некоторого двоичного слова длины s. Каждое слово длины k в исходном алфавите превратится после замены каждой его буквы на ее двоичный код в свой *двоичный образ*, а именно, в двоичное слово, имеющее длину sk. При этом исходное слово легко восстанавливается по его двоичному образу.

Пример. Алфавит, достаточный для записи сообщений на русском языке, должен включать 33 прописные и 33 строчные русские буквы, 10 цифр, знаки препинания, знаки номера, параграфа и т.д. Можно надеяться, в этом алфавите будет менее, чем 2^7 букв. Поэтому каждую из этих букв можно закодировать посредством 7-буквенного двоичного слова. Пусть, например, двоичные слова 0000001, 0011010, 1000001, 1000110, 1111111 служат, соответственно, кодами для букв а, м, я, 4, #. Тогда для выражения “4#мая” его двоичным образом будет слово 10001101111111001101000000011000001.

Начиная с этого места изложения примем, что и Алфавит Сообщений, и Алфавит Шифрограмм являются каждый двоичным алфавитом, так что все сообщения и все шифрограммы суть двоичные слова.

2.3. Переход к числам

Числа в нашей статье будут только целые, как правило неотрицательные.

Двоичное слово 10 служит записью числа два в двоичной системе счисления, короче, двоичной записью числа два. А слово 101 служит двоичной записью числа пять. Будем рассматривать только двоичные слова, начинающиеся с единицы. Каждое из них есть двоичная запись какого-либо положительного числа, и мы отождествим эти слова с теми числами, записями которых они служат. Теперь все сообщения и все шифрограммы сделались положительными числами.

Замечание. А как же все-таки быть, если сообщение, закодированное в соответствии с п. 2.2., начиналось с 0? Простейший способ разрешения проблемы таков: после создания двоичного образа надо к этому образу приписать спереди 1.

Мы совершили простую, но важную процедуру: арифметизацию ситуации. Теперь функции E и D становятся числовыми функциями, аргументами и значениями которых служат положительные целые числа. А сама ситуация выглядит так. Отправитель располагает достаточно большим запасом чисел, называемых сообщениями. На множестве этих чисел-сообщений определена функция E с числовыми же значениями. Выбрав сообщение M, Отправитель направляет Получателю число $E(M)$. Получатель, пользуясь функцией D, находит исходное M в качестве $D(E(M))$.

Коль скоро сообщения стали числами, мы можем интересоваться их числовыми свойствами, например, четностью или нечетностью или, вообще, делимостью.

Пусть даны неотрицательное a и положительное b; тогда a можно и притом единственным образом представить в виде $a = bq + r$, где q и r суть числа и $0 \leq r < b$. Процедура такого представления называется *делением с остатком числа a на число b*, а числа q и r – *неполным частным* и *остатком*. Остатком может оказаться любое число от 0 до $b - 1$. Если этот остаток r равен нулю, то неполное частное называют *полным частным,* или просто *частным*.

3. СИСТЕМА РША

Напомним, что два числа называются *взаимно простыми*, если их наибольший общий делитель равен единице. В системе РША допустимыми сообщениями служат положительные числа, меньшие некоторого весьма большого числа n, публично

объявляемого Получателем, и являющиеся взаимно простыми с этим n. Это число n будем именовать *модулем системы РША*. Модуль системы является одним из 2 параметров, задающих открытый ключ.

Здесь у читателя может возникнуть законное неудовольствие. Кажется естественным, что множество возможных сообщений не должно зависеть от правил шифрования, а тем самым и от модуля системы. Ведь это множество определяется запасом тех сведений, в передаче которых может возникнуть нужда. Мы уже поняли, что сведения эти могут кодироваться числами. Но множество этих чисел-сообщений должно, по логике вещей, возникать до шифровального ключа, а не управляться им. Мы уже объявили сообщением всякое число, которое обладает одновременно следующими двумя свойствами: (1) оно меньше модуля; (2) оно взаимно просто с ним. Первое из названных свойств не противоречит той содержательной картине, математическую модель которой мы здесь описываем. Действительно, разумно считать, что реальными сообщениями могут служить не произвольные слова в Алфавите Сообщений, но лишь слова некоторой ограниченной заранее длины. А тогда при переходе к числам, описанном в п. 2.3., все сообщения оказываются числами, не превосходящими некоторого фиксированного ограничителя m. И если только модуль системы достаточно велик, а именно, не меньше, чем m, то каждое число-сообщение, которое может возникнуть в реальности, оказывается удовлетворяющим свойству (1). Хуже обстоит дело со свойством (2): ведь нет оснований ожидать, что подлежащая передаче информация, записанная в виде слова и затем подвергнутая арифметизации, непременно приведет к числу, являющемуся взаимно простым с модулем системы. В то же время именно требование взаимной простоты с модулем обеспечивает возможность дешифровки. Как же быть? Здесь возможны два ответа. Ответ первый: в подавляющем большинстве случаев возникающее число-сообщение окажется взаимно простым с модулем. Однако, сколь бы редки ни были исключения, они все же существуют. И если передаваемое сообщение окажется числом, не взаимно простым с модулем, его шифрограмму невозможно будет расшифровать. Поэтому для тех, кто не готов соглашаться даже на самый ничтожный риск, в п. 6.3. будет предложен второй ответ.

Система РША функционирует следующим образом.

Получатель публикует в открытой печати, во-первых, модуль системы n, а, во-вторых, некоторое число e, которое назовем *открытым показателем*. Шифровальный ключ E так задается при помощи этих двух параметров: чтобы найти $E(M)$, надлежит возвести M в степень e и затем взять остаток от деления M^e на n; этот остаток и есть $E(M)$.

А Получатель, получив шифрограмму C, поступает так. Он возводит C в одному ему известную степень d и берет остаток от деления результата на n; это и есть M. Таким образом, $D(C)$ есть остаток от деления на n числа C^d. Число d будем называть *закрытым показателем*. Разумеется, как публикуемые Получателем модуль n и открытый показатель e, так и хранимый в тайне закрытый показатель d должны быть связаны некоторыми соотношениями. Более того, Получатель должен хранить в тайне некоторую дополнительную информацию, материализованную в виде пары простых чисел.

Напомним, что число называется *простым*, если оно больше единицы и не имеет делителей, отличных от единицы и самого себя. По теореме Евклида [3, Кн. IX, Предл. 20], простых чисел бесконечно много; вот начало их ряда: 2, 3, 5, 7, 11, 13, 17, 19, ...

В окончательном оформлении система РША выглядит так. Получатель сперва выбирает два очень больших (содержащих каждое сотни знаков в своей десятичной записи) простых чисел p и q. Возможность выбора сколь угодно больших простых чисел гарантируется теоремой Евклида. Произведение pq Получатель публикует в качестве модуля системы. Сами же множители p и q Получатель держит в тайне! Затем он случайным образом выбирает некоторое положительное e, являющееся взаимно простым с произведением $(p-1)(q-1)$, и публикует это e в качестве открытого показателя. Наконец, Получатель находит положительное d, удовлетворяющее тому условию, что остаток от деления произведения ed на произведение $(p-1)(q-1)$ равен 1. Это d Получатель хранит в тайне в качестве закрытого показателя.

Мы видим, что центральную роль играет то обстоятельство, что закрытый показатель известен только Получателю и более никому. Здесь может возникнуть законный вопрос. Если Получатель умеет, зная e, найти требуемое d, то ведь это же может сделать и любой другой! Нет, ответим мы, для этого надо знать p и q: вспомним требования, предъявляемые к d. Прекрасно, возразит читатель, любой может узнать p и q, коль скоро опубликовано их произведение: достаточно разложить это произведение на простые множители. Вот в этом месте и кроется главная хитрость системы РША. Дело в том, что разложить очень большое число на множители весьма и весьма сложно. Даже если мы – как в данном случае – заведомо знаем, что число является произведением ровно двух простых множителей. Таким образом, не зная p и q, а зная, только их произведение, найти p и q практически невозможно. Это утверждение о практической невозможности не является, конечно, строгой математической теоремой, а скорее экспериментальным фактом. Просто все известные к этому времени алгоритмы разложения чисел на простые множители не позволяют в обозримое время найти множители p и q по их произведению (в предложении, что эти p и q достаточно велики).

Придирчивый читатель не успокоится и здесь. Ведь если мы говорим о вычислениях, совершаемых в “обозримое время”, то можно ли считать таковыми возведение чисел M и C в степени e и d? Ответ таков. Нас интересуют не сами числа M^e и C^d, которые действительно могут оказаться “сверхбольшими”, а лишь их остатки от деления на n. Для получения же указанных остатков нет нужды вычислять сами M^e и C^d полностью: на каждом из промежуточных этапов вычисления достаточно довольствоваться остатком от деления на n соответствующего промежуточного результата.

Но тут уже начинается чистая математика, к каковой мы и переходим. Эта математика должна обосновать систему РША.

4. КРАТКОЕ ТЕОРЕТИКО-ЧИСЛОВОЕ ОБОСНОВАНИЕ

Все термины и обозначения этого параграфа будут разъяснены в п. 5.

По теореме Эйлера $M^{\phi(n)} \equiv 1 \pmod n$, где M – сообщение, а $\phi(n)$ – функция Эйлера. В нашем случае $n = pq$ и потому $\phi(n) = (p-1)(q-1)$. Поэтому

$$C^d \equiv M^{ed} \equiv M^{1+k(p-1)(q-1)} \equiv M(M^{(p-1)(q-1)})^k \equiv M \pmod n. \quad (*)$$

5. ПОДРОБНОЕ ТЕОРЕТИКО-ЧИСЛОВОЕ РАЗЪЯСНЕНИЕ

Числа n, p, q, e, d имеют указанный выше смысл.

5.1. Сравнения

Если разность чисел (возможно, отрицательных) a и b делится на положительное m, то говорят, что a и b *сравнимы по модулю* m, и выражают это на письме в форме так называемого *сравнения* $a \equiv b \pmod m$. Например, $16 \equiv 41 \pmod 5$.

Замечание. Два неотрицательных числа сравнимы по модулю m тогда и только тогда, когда совпадают их остатки от деления на m; проверку этого почти очевидного факта предоставим читателю.

По условию, остаток от деления произведения открытого и закрытого показателей на $(p-1)(q-1)$ равен 1. В силу сделанного замечания, этот факт так записывается на языке сравнений:

$$ed \equiv 1 \pmod{(p-1)(q-1)}.$$

5.2. Операции над сравнениями

Будем рассматривать сравнения по какому-либо фиксированному модулю m. Читатель легко проверит, что сравнения можно почленно умножать на число: если $a \equiv b \pmod m$, то $ak \equiv bk \pmod m$. Их можно также почленно складывать и умножать. Из возможности умножать следует возможность возводить в степень: если $a \equiv b \pmod m$, то $a^k \equiv b^k \pmod m$.

5.3. Алгоритм Евклида

Этот алгоритм описан (в геометрической форме) Евклидом в [3, Кн. VII, Предл. 2]. Он используется для нахождения наибольшего общего делителя (н.о.д.) двух чисел и состоит в следующем. Чтобы найти н.о.д. чисел a_0 и a_1, производят последовательные деления с остатком:

$$a_0 = a_1q_1 + a_2,\ a_1 = a_2q_2 + a_3,\ \ldots,\ a_{i-1} = a_iq_i + a_{i+1},\ \ldots,$$
$$\ldots,\ a_{k-2} = a_{k-1}q_{k-1} + a_k,\ a_{k-1} = a_kq_k.$$

Процесс непременно закончится, и число a_k будет искомым н.о.д.

Число c называется *линейной комбинацией* (л.к.) чисел a и b, коль скоро c можно представить в виде $ax + by$, где x и y суть целые (возможно, отрицательные!) числа. Применение алгоритма Евклида позволяет выразить н.о.д. чисел a_0 и a_1 в виде линейной комбинации этих чисел. Действительно, перепишем каждую из возникающих строк в форме разности:

$$a_2 = a_0 - a_1q_1, \quad a_3 = a_1 - a_2q_2, \text{ и т.д.}$$

Первая строка дает a_2 как л.к. исходных чисел. Подставляя это выражение во вторую строку, получаем a_3 как л.к. исходных чисел; эту л.к. подставляем в третью строку. И т.д. Таким способом последовательно выражаем каждое из чисел a_i, в том числе н.о.д. a_k, в виде л.к. исходных чисел.

5.4. Построение закрытого показателя

Пусть $m = (p-1)(q-1)$. По условию, открытый показатель e взаимно прост с m, так что н.о.д. чисел e и m равен 1. Пользуясь методом п. 5.3., представляем 1 в виде линейной комбинации чисел e и m. Тем самым находим такие целые числа x и y, что $ex + my = 1$. Далее находим такое s, что число $x + ms$ будет положительным. Поскольку $e(x + ms) = 1 - my + mse$, то остаток от деления числа $e(x + ms)$ на m будет равным 1. Поэтому $x + ms$ удовлетворяет условию, налагаемому на закрытый показатель, и может быть взято в качестве такового.

5.5. Функция Эйлера

Функция Эйлера $\phi(n)$ определяется как количество положительных чисел, меньших, чем n, и взаимно простых с n. Например, $\phi(14) = 6$, так как числа, меньшие 14 и взаимно простые с 14, суть 1, 3, 5, 9, 11, 13. Коль скоро число p простое, то все числа 1, 2, ..., $(p-1)$ взаимно просты с p, поэтому $\phi(p) = p - 1$. Можно доказать, что функция Эйлера мультипликативна; это значит, что $\phi(ab) = \phi(a)\phi(b)$ при условии, что a и b взаимно просты. Отсюда для простых p и q имеем $\phi(pq) = (p-1)(q-1)$, как утверждалось в п. 4. Впрочем, функцию Эйлера от произведения двух простых чисел нетрудно вычислить и непосредственно.

5.6. Теорема Эйлера

Пусть m и z взаимно просты. Тогда $z^{\phi(m)} \equiv 1 \pmod m$.

Ограничения на объем не позволяют нам привести здесь доказательства этой теоремы и других простейших фактов теории чисел. За такими доказательствами мы отсылаем читателя к стандартным руководствам, например, к [1] и [2].

5.7. Почему срабатывает предложенный способ дешифровки

Итак, надо убедиться, что остаток от деления C^d на n есть M. Но этот факт, в силу замечания из п. 5.1., выражается формулой (∗) из п. 4. Поэтому достаточно проверить эту формулу. Для некоторого k выполняется соотношение $ed = 1 + k(p-1)(q-1)$. Фиксируем это k. Первые три входящие в формулу сравнения суть равенства. Для обоснования заключительного сравнения применяем теорему Эйлера, беря в ее формулировке из п. 5.6. в качестве модуля сравнения m модуль n системы РША, а в качестве z – сообщение M; мы вправе применить теорему к этим объектам, поскольку, по условию, сообщение и модуль системы взаимно просты. Согласно п. 5.5. $\phi(m)$ в данном случае равно $(p-1)(q-1)$. Поэтому в нашем случае теорема Эйлера будет выглядеть так:

$$M^{(p-1)(q-1)} \equiv 1 \pmod{pq}.$$

Далее мы пользуемся возможностью почленно возводить сравнения в степень (п. 5.2.) и получаем

$$(M^{(p-1)(q-1)})^k \equiv 1 \pmod{pq}.$$

Наконец, мы умножаем обе части сравнения на M и, помня, что n есть pq, получаем в точности заключительное сравнение п. 4.

6. ОБСУЖДЕНИЕ

6.1. Как Отправитель и Получатель могут облегчить себе шифровку и дешифровку

В конце п. 3 было замечено, что коль скоро нас интересуют не сами числа M^e и C^d, а лишь их остатки от деления на n, нет нужды вычислять целиком указанные “сверхбольшие” числа.

Пусть $[X]$ обозначает остаток от деления числа X на n. Очевидно, $[X] \equiv X \pmod n$. Сравнения можно умножать, а потому $[AB] \equiv AB \equiv [A][B] \equiv [[A][B]] \pmod n$. Далее, если $[X] \equiv [Y] \pmod n$, то $[X] = [Y]$. Поэтому $[AB] = [[A][B]]$. Таким образом, чтобы получить остаток от деления на n произведения чисел A и B, можно сперва найти остатки $[A]$ и $[B]$, а затем взять остаток от деления на n произведения этих остатков. Поэтому для получения, скажем, остатка $[M^{11}]$ разумно произвести следующие вычисления: $[M^2] = [[M][M]]$, $[M^4] = [[M^2][M^2]]$, $[M^8] = [[M^4][M^4]]$, $[M^{11}] = [[[M][M^2]][M^8]]$. Изложим эту методику более формально и для общего случая. Мы будем говорить лишь о вычислении $[M^e]$, но все сохраняет силу и для вычисления $[C^d]$. Чтобы быстро вычислить $[M^e]$, найдем двоичное представление числа e, то есть такие числа $k, b_0, b_1, \ldots, b_k$, что

$$e = b_0 + b_1 \cdot 2 + \ldots + b_i \cdot 2^i + \ldots + b_k \cdot 2^k,$$
$$b_0, b_1, \ldots, b_k = 0, 1.$$

Затем последовательно вычислим

$$[M^2], [M^4], \ldots, [M^{2^i}], \ldots, [M^{2^k}].$$

После этого перемножим (по модулю n, то есть заменяя каждый раз получаемый результат его остатком от деления на n) числа $[M^{2^i}]$ для тех i, для которых $b_i = 1$. Полученное число, очевидно, равно $[M^e]$. Ясно также, что количество выполненных операций не превосходит $2k \le 2\log_2 n$.

6.2. Некоторые меры предосторожности

Если $M^e < n$, секретность переписки оказывается под угрозой. В этом случае шифрограмма C совпадает с M^e. А тогда перехвативший шифрограмму злоумышленник может восстановить исходное сообщение M путем извлечения корня степени e из шифрограммы. Хотя злоумышленник не знает заранее, совпадает ли C с числом M^e, он может выдвинуть гипотезу о таком совпадении. А критерием, на основании которого он может судить о верности гипотезы, служит выполнение равенства $E(\sqrt[e]{C}) = C$, что легко проверяется. Поэтому сообщение M, рассматриваемое как число, не должно быть слишком мало. Что же делать, если надо передать небольшое сообщение? Расширим алфавит сообщений за счет специального разделительного знака, например, звездочки ∗. Малость сообщения как числа равносильна краткости сообщения как слова. В таком случае мы поставим в конце сообщения знак ∗, а после этого знака выпишем достаточно длинную и беспорядочную (случайную, как говорят математики) последовательность букв. Мы получим новое, длинное сообщение, значимым в котором будет только начальная часть, предшествующая звездочке. Это новое сообщение будет уже кодироваться достаточно большим числом.

6.3. Что делать, если число-сообщение не взаимно просто с модулем системы

Такие случаи практически почти никогда не будут встречаться. Ведь простые числа p и q, дающие в произведении модуль системы n, астрономически велики. А тогда отношение $(p-1)(q-1)/n$, показывающее долю чисел, взаимно простых с модулем, среди всех чисел, не превосходящих модуля, практически неотличимо от единицы. Однако можно предложить план действий, сводящий риск до абсолютного нуля. Из любых трех последовательно идущих чисел хотя бы одно взаимно просто с модулем. А потому допустим следующий прием. Отправитель, желая передать число a, образует три числа $3a$, $3a-1$, $3a-2$. По меньшей мере одно из них (а с высочайшей

вероятностью все три!) будет взаимно просто с модулем. Далее Отправитель может действовать одним из двух способов.

Первый способ. Отправитель, пользуясь алгоритмом Евклида, проверяет три построенные числа на взаимную простоту с модулем и, как только обнаруживает взаимно простое число, шифрует его по методу РША и отправляет шифрограмму Получателю. Получатель, дешифруя шифрограмму, находит число вида $3a - x$, где x есть 0, 1 или 2, и далее восстанавливает a.

Второй способ. Отправитель шифрует все три числа и отправляет Получателю три шифрограммы. Теперь уже Получатель производит проверку шифрограмм на взаимную простоту с модулем. Обнаружив таковую, он дешифрует ее и восстанавливает a. (Здесь мы используем тот факт, что в Системе РША сообщение и его шифрограмма либо одновременно взаимно просты, либо одновременно не взаимно просты с модулем.)

Оба способа отличаются друг от друга по существу лишь тем, на кого возлагается бремя проверки взаимной простоты. Существенно другое: количество допустимых сообщений теперь уменьшается втрое.

6.4. Заключительное замечание

Конечно, никогда не может быть стопроцентной уверенности, что злоумышленник не сможет расшифровать перехваченную им шифрограмму. Положение дел здесь примерно такое же, как в случае с цифровым замком в автоматической камере хранения: ведь и там не исключено, что преступник найдет требуемую комбинацию путем догадки (поэтому и не рекомендуется в качестве цифровых комбинаций брать нечто осмысленное, например, те или иные даты). И в случае автоматической камеры хранения, и в случае системы тайнописи РША речь может идти только о практической невозможности, лучше сказать, чрезвычайно малой вероятности расшифровки (настолько малой, что ею можно пренебречь).

ЛИТЕРАТУРА

1. *Виноградов И.М.* Основы теории чисел. М.: Физматлит, 1965. Изд. 7-е, исправл.

2. *Дэвенпорт Г.* Высшая арифметика. Введение в теорию чисел. Пер. с англ. М.: Физматлит, 1965.

3. Начала Евклида. Книги VII – X / Пер. с греч. М.-Л.: Гостехиздат, 1949.

4. *Rivest R.L.*, *Shamir A.*, *Adleman L.M.* A method for obtaining digital signatures and public-key cryptosystems // Communications ACM, 1978. V. 21. PP. 120–126.

5. *Rivest R.L.* Cryptography / Handbook of Theoretical Computer Science. Vol. A. Algorithms and Complexity / J. van Leeuwen, ed. Amsterdam: Elsevier; Cambridge, Mass.: The MIT Press., 1990. PP. 717 – 755.

* * *

Владимир Андреевич Успенский, доктор физико-математических наук, профессор, зав. кафедрой математической логики и теории алгоритмов Московского государственного университета им. М.В. Ломоносова. Автор 10 книг и 90 научных статей.