Государственный Университет – Высшая Школа Экономики

ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ АЛГОРИТМОВ ДИСКРЕТНОГО ЛОГАРИФМИРОВАНИЯ, ИСПОЛЬЗУЮЩИХ ФАКТОРНУЮ БАЗУ

А.А. Савельева

Москва 2007

Содержание

Введение

Проблема защиты информационных ресурсов в настоящее время приобретает все большее значение. Так, по данным отчета CSI/FBI Computer Crime and Security Survey 2006 [7], средний ущерб каждой компании, в которой в минувшем году было зафиксировано нарушение информационной безопасности, составил $167,713. По некоторым оценкам, экономические потери от злонамеренных атак на банковские системы по всему миру составляют ежегодно около 130 млрд. долларов. Выбор необходимой степени защиты информации и средств ее обеспечения является важной задачей и должен учитывать ряд параметров [16]: уровень секретности информации; ее стоимость; время, в течение которого она должна оставаться в тайне и т.д.

Обычно выделяют следующие методы защиты информации от умышленных деструктивных воздействий [25]:

- методы обеспечения физической безопасности компонентов системы;
- ограничение доступа;
- разграничение доступа;
- разделение доступа (привилегий) – разрешение доступа только при одновременном предъявлении полномочий всех членов группы;
- криптографическое преобразование информации и реализованные на его основе криптографические протоколы.

Целью данной работы является анализ надежности алгоритмов, осуществляющих криптографическое преобразование информации.

На сегодняшний день на платформах Windows XP и Windows Server 2003 компания Microsoft рекомендует использовать следующие криптографические алгоритмы [8]:

- AES-128 (или AES-192, или AES-256)

- RSA 2048 (или с еще более длинным ключом)
- “SHA-2” (т.е. SHA-256 или SHA-512)
- DSA (или SHA-2/RSA)

Криптография Windows Vista (и Longhorn Server) соответствует рекомендациям Агентства Национальной Безопасности США и Национального института стандартов и технологии (NIST) по реализации протоколов “Suite-B” [8] и предусматривает использование асимметричных криптоалгоритмов на основе эллиптических кривых. Алгоритмы “Suite-B” включают:

- AES (шифрование)
- EC-DSA (электронно-цифровая подпись)
- EC-DH или EC-MQV (обмен секретными ключами)
- SHA-2 (хеширование)

В настоящее время общепризнанным является подразделение криптографических алгоритмов на следующие основные категории:

- алгоритмы шифрования с секретным ключом (симметричные)
    - блочные шифры
    - поточные шифры
- алгоритмы шифрования с открытым ключом (асимметричные)

AES относится к категории симметричных шифров. Остальные перечисленные алгоритмы используются для реализации криптосистем с открытым ключом. В данной работе мы ограничимся исследованием стойкости асимметричных шифров.

В асимметричной криптографии для зашифрования и расшифрования используются различные функции. Асимметричные алгоритмы основаны на ряде математических проблем, на которых и базируется их стойкость. Применение алгоритмов шифрования с открытым ключом позволяет:

- избавиться от необходимости секретных каналов связи для предварительного обмена ключами;
- свести проблему взлома шифра к решению трудной математической задачи, т.е., в конечном счете, принципиально по-другому подойти к обоснованию стойкости криптосистемы;
- решать средствами криптографии задачи, отличные от шифрования, например, задачу обеспечения юридической значимости электронных документов.

Подтверждение авторства сообщений может осуществляться при помощи криптографических средств, что абсолютно необходимо для дистанционного управления ресурсами. Лицо, которое управляет чьими-либо ресурсами, должно обладать возможностью доказать, что выполненное им распоряжение было получено именно от владельца. Эта задача приобрела особенную актуальность с появлением электронной коммерции: в качестве ресурса в данном случае выступают деньги на банковском счету владельца.

Описанную проблему позволяют решить различные схемы электронно-цифровой подписи (ЭЦП). Любая схема ЭЦП обязана определить три следующих алгоритма:

- алгоритм генерации ключевой пары для подписи и ее проверки;
- алгоритм подписи;
- алгоритм проверки подписи.

RSA [10] – криптографическая система с открытым ключом, обеспечивающая оба механизма защиты: шифрование и цифровую подпись. Криптосистема RSA была разработана в 1977 году и названа в честь авторов: Рональда Ривеста, Ади Шамира и Леонарда Адлемана.

Принцип её действия в следующем. Берутся два больших случайных простых числа p и q приблизительно равной разрядности и вычисляется их произведение $n = p \cdot q$. Затем выбирается число e, взаимно простое с произведением $(p-1)\cdot(q-1)$ и вычисляется число $d = e^{-1}\left(\mathrm{mod}(p-1)\cdot(q-1)\right)$, взаимно простое с n.

Числа e и n становятся открытым ключом, число d – закрытым. Чтобы создать шифртекст c, отправитель возводит сообщение m в степень e по модулю n, где e и n – показатели открытого ключа получателя: $c = m^e (\mathrm{mod}\, n)$.

Чтобы расшифровать полученный шифртекст c, получатель вычисляет c в степени d по модулю n: $m = c^d (\mathrm{mod}\, n)$.

Если абонент А хочет подтвердить свое авторство сообщения, он сначала шифрует его на своем секретном ключе, а потом на открытом ключе абонента Б. Соответственно, абонент Б применяет к полученному сообщению свой секретный ключ и открытый ключ абонента А; успешное расшифрование является гарантией того, что отправить сообщение мог только абонент А.

Преимуществом алгоритма RSA, рекомендуемого к использованию в Windows XP/Server 2003, является хорошая изученность и отсутствие на сегодняшний день алгоритмов факторизации с полиномиальной сложностью.

Схема Эль-Гамаля [3] основана на трудности вычисления дискретных логарифмов в конечном поле в сравнении с лёгкостью возведения в степень в том же самом поле.

Для генерации пары ключей сначала выбирается простое число p и два случайных числа, g и x; оба эти числа должны быть меньше p. Затем вычисляется $y = g^x (\mathrm{mod}\, p)$.

Открытым ключом становятся y, g и p. И g, и p можно сделать общими для группы пользователей. Закрытым ключом является x. Теперь, чтобы зашифровать сообщение m, сначала выбирается случайное k, взаимно простое с $p-1$. Затем вычисляются $a = g^k \pmod p$, $b = y^k \cdot m \pmod p$. Пара a и b является шифртекстом, что увеличивает исходное сообщение в два раза. Для расшифрования вычисляется $m = b / a^x \pmod p$.

DSA (Digital Signature Algorithm) - алгоритм цифровой подписи, принятый в 1994 году в качестве стандарта США на ЭЦП [4] и действовавший до 2001 г., представляет собой один из вариантов схемы Эль-Гамаля.

Национальный институт стандартов и технологии NIST, вместе с АНБ, разработал алгоритм SHA [4] (Secure Hash Algorithm – алгоритм стойкого хеширования), требуемый для обеспечения стойкости алгоритма цифровой подписи DSA. По предположениям Microsoft, алгоритм SHA-2, предусматривающий генерацию хеш-значений длиной 224, 256, 384 и 512 разрядов, еще на протяжение нескольких лет будет оставаться стойким, но в конечном счете будет заменен.

Ряд успешных атак, описанных, например, в [9], на системы, основанные на сложности дискретного логарифмирования в конечных полях, привел к тому, что стандарты ЭЦП России [23] и США [4] в 2001 году были обновлены: переведены на эллиптические кривые [22, 5]. Алгоритмы ЭЦП при этом не изменились, однако вместо элементов конечного поля $GF(2^n)$ или $GF(p)$ они оперируют эллиптическими числами, т.е. решениями уравнения эллиптических кривых над указанными конечными полями, а роль операции возведения в степень в конечном поле выполняет операция взятия кратной точки эллиптической кривой.

Специальный выбор типа эллиптической кривой позволяет не только во много раз усложнить задачу взлома схемы ЭЦП, но и уменьшить рабочий размер блоков данных. Старый российский стандарт ЭЦП оперирует 1024-

битовыми блоками, а новый, основанный на эллиптических кривых, — 256-битовыми, и при этом обладает большей стойкостью.

EC-DSA (Elliptic Curve Digital Signature Algorithm) – это классический алгоритм DSA, «переведенный» на эллиптические кривые. Стандарт FIPS 186-2 предусматривает 256- и 284- битный рабочий размер блоков данных, но Microsoft также поддерживает 521-разрядные ключи.

Как известно [12], далеко не все присутствующие на рынке криптографические средства обеспечивают обещанный уровень защиты. Важность этой проблемы подчеркивается и в работе [30]. Системы и средства защиты информации (СЗИ) характеризуются тем, что для них не существует простых и однозначных тестов, позволяющих убедиться в надежной защите информации. Например, чтобы проверить работоспособность системы связи, достаточно провести ее испытания. Тем не менее, успешное завершение этих испытаний не дает возможность сделать вывод, что встроенная подсистема защиты информации тоже является работоспособной. Часто задача определения эффективности СЗИ при использовании криптографических методов защиты оказывается даже более трудоемкой, чем разработка СЗИ, т.к. требует наличия специальных знаний и более высокой квалификации, чем задача разработки. Как правило, анализ нового криптографического алгоритма является новой научной, а не инженерной задачей.

Повышение производительности вычислительной техники и появление новых видов атак на шифры ведет к понижению стойкости известных криптографических алгоритмов. Для уменьшения возможного ущерба, вызванного несвоевременной заменой потерявшего свою стойкость криптоалгоритма, необходима периодическая перепроверка стойкости криптоалгоритмов, которая включает в себя как разработку новых методов криптоанализа, так и повышение эффективности существующих методов [15].

То обстоятельство, что любую задачу отыскания способа раскрытия некоторой конкретной криптосистемы можно переформулировать как

привлекательную математическую задачу, при решении которой удается использовать многие методы той же теории сложности, теории чисел и алгебры, привело к раскрытию многих криптосистем. Практически все используемые алгоритмы асимметричной криптографии основаны задачах факторизации и дискретного логарифмирования в различных алгебраических структурах. Задача дискретного логарифмирования считается более сложной с алгоритмической точки зрения; если будет найден полиномиальный алгоритм ее решения, станет возможным и разложение на множители (обратное не доказано).

Объектом исследования данной работы стали алгоритмы дискретного логарифмирования и методы повышения их эффективности.

Криптоанализ систем шифрования, основанных на сложности задачи дискретного логарифмирования

Наиболее эффективные на сегодняшний день алгоритмы дискретного логарифмирования имеют уже не экспоненциальную, а субэкспоненциальную временную сложность. Это алгоритмы “index-calculus”, использующие факторную базу. Первый такой алгоритм был предложен Адлеманом [1] и имеет временную сложность $L_p\left[\frac{1}{2}; c\right]$ при вычислении дискретного логарифма в простом поле $\mathbb{Z}_p$ ($L_N[\gamma; c] = e^{(c+o(1))(\log N)^{\gamma}(\log\log N)^{1-\gamma}}$, где $0 < \gamma < 1, c = const, c > 0$). Идея использования факторной базы применялась и ранее, например, в [13]. На практике алгоритм [1] оказался недостаточно эффективным; Копперсмит, Одлыжко и Шреппель предложили алгоритм дискретного логарифмирования COS [2] с эвристической оценкой сложности $L_p\left[\frac{1}{2}; 1\right]$ операций. Алгоритм решета числового поля [11], предложенный Широкауэром, при $p > 10^{100}$

работает эффективнее различных модификаций метода COS; его временная сложность составляет порядка $L_p\left[\frac{1}{3};(64/9)^{1/3}\right]$ арифметических операций.

Пусть G - мультипликативная абелева группа. Вычислить дискретный логарифм b по основанию a в группе G означает найти $x \in G$, при котором $a^x = b$. Свойства дискретного логарифма во многом схожи со свойствами обычного логарифма в поле действительных чисел. Например, выполняется тождество $\log_a(h \cdot j) \equiv \log_a(h) + \log_a(j) \pmod{|G|}$, где $|G|$ - порядок группы, a - образующая.

Основная идея методов “index-calculus” заключается в том, что если

$$\prod_{i=1}^{m} x_i = \prod_{j=1}^{n} y_j$$

для некоторых элементов конечного поля $\mathbb{Z}_p$, то

$$\sum_{i=1}^{m} \log_a x_i \equiv \sum_{j=1}^{n} \log_a y_j \pmod{p-1} \qquad (1)$$

Получив достаточно много соотношений (1) (причем хотя бы одно из них должно включать элемент g, для которого $\log_a g$ известен), можно решить систему линейных уравнений относительно неизвестных $\log_a x_i$ и $\log_a y_j$ в кольце вычетов $\mathbb{Z}_{p-1}$ при условии, что количество неизвестных в уравнениях не слишком велико.

Самый простой подход к генерации соотношений вида (1) – выбрать произвольный элемент $g \in \mathbb{Z}_p$, вычислить $u = a^g \pmod{p}$ и с помощью некоторого перебора попытаться найти числа, удовлетворяющие соотношению:

$$u = \prod p_i, \qquad (2)$$

где p_i - простые числа, удовлетворяющие соотношению $p_i < B$ для некоторой границы B. Если соотношение выполняется, то u - гладкий элемент с границей гладкости B. Элемент является *гладким* [9], если существует множество небольших элементов, через которые его можно выразить с помощью простого алгоритма.

Выделяются два основных этапа в работе алгоритмов: на первой, подготовительной стадии, формируется факторная база и на ее основе генерируется система линейных уравнений в кольце $\mathbb{Z}_{p-1}$; вид факторной базы (множество простых чисел, неприводимых многочленов или других объектов) и способы получения матрицы системы зависят от выбранного алгоритма. На второй стадии (которая является общей для рассматриваемых алгоритмов) требуется получить решение этой системы. Интенсивные предварительные вычисления для каждого поля достаточно выполнить только один раз. Затем можно быстро вычислять различные дискретные логарифмы.

Алгоритмов, осуществляющих дискретной логарифмирование на эллиптических кривых в общем случае хотя бы с субэкспоненциальной сложностью, на сегодняшний день не существует. Тем не менее, в одной из работ И.А. Семаева [36] рассматривается метод, идейно близкий алгоритмам Адлемана дискретного логарифмирования в конечном поле [1]. В другой работе для эллиптических кривых специального вида (на модуль арифметики и на мощность группы точек накладываются некоторые условия) И.А. Семаев указывет способ сведения задачи логарифмирования в группе точек эллиптической кривой к задаче логарифмирования в некотором расширении простого поля, причем сведение осуществляется с полиномиальной сложностью с использованием так называемого спаривания Вейля [29], после чего можно использовать известные субэкспоненциальные методы. В работе [6], опубликованной за рубежом, получены аналогичные результаты.

Таким образом, все субэкспоненциальные методы дискретного логарифмирования сводятся к задаче решения систем линейных уравнений в кольцах вычетов.

Методы решения систем линейных уравнений в кольцах вычетов

Обзор существующих методов решения систем линейных уравнений в кольцах вычетов

Анализ методов решения систем линейных уравнений в кольцах вычетов, описанных в современной литературе, выявил ряд недостатков, которые затрудняют использование этих алгоритмов на практике.

В монографии [20] задача сводится к решению систем линейных уравнений над полями Галуа. Пусть

$$\sum_{j=1}^{m} a_{ij} x_j \equiv b_i (\mathrm{mod}\, p), \qquad i = \overline{1,n} \qquad (3)$$

где $p = \prod_{k=1}^{t} q_k^{\alpha_k}$, тогда решение системы (3) сводится к решению семейства систем

$$\sum_{j=1}^{m} a_{ij} x_j \equiv b_i (\mathrm{mod}\, q_k^{\alpha_k}), \qquad i = \overline{1,n}, \quad k = \overline{1,t} \qquad (4)$$

где неизвестные значения $x_j (\mathrm{mod}\, q_k^{\alpha_k})$ для фиксированного k представляются в виде

$$x_j \equiv x_{j0} + x_{j1} q_k + \ldots + x_{j,\alpha_k - 1} q_k^{\alpha_k - 1} \ (\mathrm{mod}\, q_k^{\alpha_k}), \qquad (5)$$

Здесь $0 \le x_{jl} \le q_k - 1, \quad l = \overline{0, \alpha_k - 1}$.

Редуцируя систему (4) к модулю q_k, получаем систему уравнений:

$$\sum_{j=1}^{m} a_{ij} x_{j0} \equiv b_i (\mathrm{mod}\, q_k), \qquad i = \overline{1,n}$$

над полем Галуа $\mathbb{Z}_{q_k}$. Если мы найдем все x_{j0}, $j=\overline{1,m}$, то, подставляя x_j в виде (5) с известными x_{j0} в систему (4), редуцируя ее к модулю q_k^2 и затем поделив на q_k, мы получим систему линейных уравнений над полем $\mathbb{Z}_{q_k}$ относительно неизвестных x_{j1}, $j=\overline{1,m}$, и т.д. В конечном счете, найдя значение $x_j(\mathrm{mod}\, q_k^{\alpha_k})$ для всех k, мы восстановим $x_j(\mathrm{mod}\, p)$ по китайской теореме об остатках (см. [Ноден]).

Рассмотрим работу метода на примере следующей системы линейных уравнений в кольце вычетов $\mathbb{Z}_{36}$:

$$\begin{cases} 26x + 3y = 4 \\ 9x + 34y = 1 \end{cases}$$

В нашем примере $p = 36 = 2^2 \cdot 3^2$, т.е. для решения одной системы в кольце вычетов придется решить 4 системы над полями Галуа. Но основным недостатком метода является необходимость разложения на множители числа p: вопрос о существовании алгоритма факторизации с полиномиальной сложностью является одной из открытых проблем современной теории чисел.

Другой метод предполагает сведение системы линейных уравнений в кольце вычетов к системе линейных диофантовых уравнений. При помощи одного из известных алгоритмов (см. [28] или [20]) расширенная матрица системы $(A\,|\,b)$ приводится к *ступенчатому виду* $(A'\,|\,b')$ и вычисляется *правая матрица перехода* R размером $(m+1)\times(m+1)$, такая, что: $(A\,|\,b)\times R = (A'\,|\,b')$. На основании матрицы R можно получить общее решение системы.

Проиллюстрируем применение данного способа на примере. Система линейных диофантовых уравнений, соответствующая приведенной выше системе в $\mathbb{Z}_{36}$, имеет вид:

$$\begin{cases} 26x + 3y + 36v_1 = 4 \\ 9x + 34y + 36v_2 = 1 \end{cases}.$$

Ее общее решение в кольце целых чисел:

$$\begin{cases} x = \ 5653025 \ + t_0 \cdot 1224 \ + t_1 \cdot (-21492) \\ y = -1496390 \ + t_0 \cdot (-324) \ + t_1 \cdot 5688 \\ v_1 = -3958042 \ + t_0 \cdot (-857) \ + t_1 \cdot 15048 \\ v_2 = \ 0 \ + t_0 \cdot 0 \ + t_1 \cdot 1 \end{cases}, \quad t_0, t_1 \in \mathbb{Z}$$

Редуцируя результат к модулю 36, получаем: $x = 17, y = 22$.

Однако при решении систем линейных уравнений в кольцах вычетов наблюдается экспоненциальный рост длины коэффициентов. Так, в нашем примере коэффициенты исходной матрицы ограничены числом 36, тогда как в целых числах мы получили общее решение с коэффициентами $\sim 10^6$. Заметим, что этот результат соответствует системе в кольце вычетов, состоящей всего лишь из двух уравнений с двумя неизвестными.

Для того, чтобы избежать экспоненциального роста длины коэффициентов, разработаны специальные методы решения систем линейных алгебраических уравнений над кольцом целых чисел, такие как модификация метода Гаусса и построение нормальной диагональной формы Смита (см. [27]). Несмотря на то, что эти алгоритмы являются полиномиальными, их сложность существенно превышает сложность алгоритма Гаусса при решении систем в полях Галуа. Так, для системы из n уравнений с n неизвестными, коэффициенты которой по абсолютной величине не превосходят α, временная сложность модифицированного алгоритма Гаусса при использовании самого быстрого алгоритма умножения составляет $O\left(n^4(\log\alpha + \log n)\right)$. Трудоемкость построения нормальной диагональной формы Смита матрицы $A_{n\times m}$, где $|a_{ij}| \le \alpha, i = \overline{1,n}, j = \overline{1,m}$, ограничена величиной $O\left(n^2 m^2 \log\alpha\right)$.

Метод решения систем линейных уравнений в кольцах вычетов [14], предлагаемый в данной работе, лишен недостатков вышеописанных алгоритмов и показал свою эффективность при программной реализации.

Описание разработанного метода

В основе разработанного метода, представляющего собой модификацию схемы Жордана, лежит преобразование строк матрицы с использованием коэффициентов Безу, которые позволяет вычислить расширенный алгоритм Евклида (см. рис.1).

Евклид(a, b)

1. $\begin{pmatrix} d & x & y \\ n & r & s \end{pmatrix} \leftarrow \begin{pmatrix} a & 1 & 0 \\ b & 0 & 1 \end{pmatrix}$
2. $c \leftarrow \lfloor d / n \rfloor$
3. $\begin{pmatrix} d & x & y \\ n & r & s \end{pmatrix} \leftarrow \begin{pmatrix} 0 & 1 \\ 1 & -c \end{pmatrix} \times \begin{pmatrix} d & x & y \\ n & r & s \end{pmatrix}$
4. *ЕСЛИ* $n > 0$
5. | *ТО* перейти к шагу 2;
6. | *ИНАЧЕ* вернуть (d, x, y, r, s)

Рис.1

В результате работы алгоритма Евклида мы получаем:

$$НОД(a, b) = d = a \cdot x + b \cdot y,$$

$$0 = n = a \cdot r + b \cdot s.$$

При $a = 26 = a_{11}$, $b = 9 = a_{21}$:

$$НОД(26, 9) = 1 = 26 \cdot (-1) + 9 \cdot (3),$$

$$0 = 26 \cdot (9) + 9 \cdot (-26).$$

Применяя к 1-й и 2-й строке расширенной матрицы нашей системы преобразования, соответствующие преобразованиям алгоритма Евклида над поступающими на его вход коэффициентами $a_{11} = 26$ и $a_{21} = 9$, в результате мы получаем матрицу, строчно эквивалентную исходной (см. [21]):

$$\begin{matrix}[1]\\[2]\end{matrix} \left(\begin{array}{cc|c} 26 & 3 & 4 \\ 9 & 34 & 1 \end{array}\right) \xrightarrow{[1]-[2]\cdot 2} \left(\begin{array}{cc|c} 8 & 7 & 2 \\ 9 & 34 & 1 \end{array}\right) \xrightarrow{[1]\leftrightarrow[2]} \left(\begin{array}{cc|c} 9 & 34 & 1 \\ 8 & 7 & 2 \end{array}\right)$$

$$\begin{matrix}[1]\\[2]\end{matrix} \left(\begin{array}{cc|c} 9 & 34 & 1 \\ 8 & 7 & 2 \end{array}\right) \xrightarrow{[1]-[2]\cdot 1} \left(\begin{array}{cc|c} 1 & 27 & 35 \\ 8 & 7 & 2 \end{array}\right) \xrightarrow{[1]\leftrightarrow[2]} \left(\begin{array}{cc|c} 8 & 7 & 2 \\ 1 & 27 & 35 \end{array}\right) \quad (6)$$

$$\begin{matrix}[1]\\[2]\end{matrix} \left(\begin{array}{cc|c} 8 & 7 & 2 \\ 1 & 27 & 35 \end{array}\right) \xrightarrow{[1]-[2]\cdot 8} \left(\begin{array}{cc|c} 0 & 7 & 10 \\ 1 & 27 & 35 \end{array}\right) \xrightarrow{[1]\leftrightarrow[2]} \left(\begin{array}{cc|c} 1 & 27 & 35 \\ 0 & 7 & 10 \end{array}\right)$$

В полученной матрице:

$$\begin{pmatrix} A(1,*) \\ A(2,*) \end{pmatrix} = \begin{pmatrix} x' & y' \\ r' & s' \end{pmatrix} \times \begin{pmatrix} A(1,*) \\ A(2,*) \end{pmatrix},$$

где x', y', r', s' удовлетворяют условию: $НОД(a_{11}, a_{21}) = a_{11} \cdot x' + a_{21} \cdot y'$, $0 = a_{11} \cdot r' + a_{21} \cdot s'$ (запись $A(k,*)$ используется для обозначения k-й строки расширенной матрицы A). Коэффициенты Безу $x' = -1, y' = 3, r' = 9, s' = -26$ можно получить, оперируя лишь коэффициентами a_{11} и a_{21}. Тогда цепь преобразований (6) сводится к одному преобразованию следующего вида:

$$\begin{matrix} [1] \\ [2] \end{matrix} \begin{pmatrix} 26 & 3 & \vert 4 \\ 9 & 34 & \vert 1 \end{pmatrix} \xrightarrow[[2]'=[1]\cdot 9+[2]\cdot 10]{[1]'=[1]\cdot 35+[2]\cdot 3} \begin{pmatrix} 1 & 27 & \vert 35 \\ 0 & 7 & \vert 10 \end{pmatrix}$$

С учетом того, что коэффициент $a_{22} = 7$ обратим в $\mathbb{Z}_{36}$: $7^{-1} \equiv 31 (\mathrm{mod}\, 36)$, преобразуем матрицу к единичной и получаем решение:

$$\begin{matrix} [1] \\ [2] \end{matrix} \begin{pmatrix} 1 & 27 & \vert 35 \\ 0 & 7 & \vert 10 \end{pmatrix} \xrightarrow[[2]'=[2]\cdot 31]{[1]'=[1]+[2]\cdot 27} \begin{pmatrix} 1 & 0 & \vert 17 \\ 0 & 1 & \vert 22 \end{pmatrix}$$

В общем виде алгоритм решения систем линейных уравнений в кольцах вычетов, представляющий собой модификацию метода Жордана, описан на рис.2. Для простоты рассмотрен случай, когда число уравнений системы равно числу неизвестных. Алгоритм легко модифицируется для решения системы, имеющей матрицу произвольного размера.

Предложенный алгоритм является корректным, т.е. полученная в результате преобразований система равносильна исходной (иначе говоря, решения системы не теряются и новые решения не появляются).

Запишем систему уравнений (3) в матричном виде:

$$Ax = b \qquad (7)$$

По теореме о равносильности систем линейных уравнений**:**

Если U - обратимая $(n \times n)$-матрица над R (R - произвольное коммутативное кольцо с единицей), тогда система уравнений (7) равносильна системе $(UA)x = Ub$.

(Доказательство см. в [21]).

Следствие из этой теоремы:

Если матрицы (A, b) *и* (C, δ) *строчно эквивалентны, то система уравнений* (7) *равносильна системе* $Cx = \delta$.

Модиф _ Жордан(A)

1. $n \leftarrow$ *Число _ Строк*(A)
2. $i \leftarrow 1$
3. *ДЛЯ* $j = \overline{i+1, n}$ *ЦИКЛ*
4. | *ВЫЧИСЛИТЬ* $x', y', r', s' : \left\{ \begin{matrix} НОД(a_{ii}, a_{ji}) = a_{ii} \cdot x' + a_{ji} \cdot y' \\ 0 = a_{ii} \cdot r' + a_{ji} \cdot s' \end{matrix} \right\}$
5. | $\begin{pmatrix} A(i,*) \\ A(j,*) \end{pmatrix} \leftarrow \begin{pmatrix} x' & y' \\ r' & s' \end{pmatrix} \times \begin{pmatrix} A(i,*) \\ A(j,*) \end{pmatrix}$
6. *ЕСЛИ* коэффициент a_{ii} необратим в $\mathbb{Z}_p$
7. | *ТО* выйти из алгоритма {*матрица вырождена*}
8. | *ИНАЧЕ* {*обнуляем все элементы* $i-$*го столбца выше ведущего*}
9. | | $A(i,*) \leftarrow A(i,*) \cdot a_{ii}^{-1}$
10. | | $A(j,*) \leftarrow A(j,*) - A(i,*) \cdot a_{ji}, \quad j = \overline{1, i-1}$
11. $i \leftarrow i+1$
12. *ЕСЛИ* $i \le n$
13. | *ТО* перейти к шагу 2;
14. | *ИНАЧЕ* вернуть(A)

Рис.2

Поскольку преобразования матрицы в описанном модифицированном методе Жордана базируются на элементарных преобразованиях строк *(элементарными преобразованиями строк матрицы* с элементами из коммутативного кольца с единицей называют (см. [24]) умножение любой ее строки на обратимый элемент кольца; прибавление к любой ее строке другой строки, умноженной на произвольный элемент кольца; транспозицию строк), то полученная на выходе алгоритма матрица строчно эквивалентна исходной (см. [21]). Тогда по приведенному выше следствию соответствующие системы уравнений являются равносильными. Что и требовалось доказать.

Предложенный алгоритм обладает временной сложностью $O\left(n\cdot(nm+\log p)\right)$ для системы в кольце вычетов по модулю p, в которой n - число уравнений системы, m - число неизвестных.

Для получения этой формулы воспользуемся оценкой временной сложности алгоритма Евклида $T(a,b)=O\left(1+\log_{\varphi}\dfrac{b}{НОД(a,b)}\right)$, где $a>b\geq 0,\ \varphi=(1+\sqrt{5})/2$ (доказательство этой оценки предлагается в [26], в качестве упражнения).

На каждом j-м шаге процедура, реализующая алгоритм Евклида, вызывается j раз: первым параметром является текущее значение ведущего элемента, в качестве второго на вход последовательно подаются a_{ij} $(i=\overline{j+1,\,n})$ и p. Пусть d_i - значение ведущего элемента на i-й итерации цикла:

$$d_0=a_{jj},\ d_1=НОД(a_{jj},\,a_{j+1,\,j})=НОД(d_0,\,a_{j+1,\,j}),\ \dots,$$
$$d_k=НОД(d_{k-1},\,a_{j+k,\,j}),\ \dots,\ d_{n-j}=НОД(d_{n-j-1},\,a_{n,\,j}).$$

Тогда число операций оценивается неравенством:

$$\sum_{i=1}^{n-j}\left(1+\log\frac{\min\left\{d_{i-1},\,a_{i,\,j}\right\}}{НОД(d_{i-1},\,a_{i,\,j})}\right)\leq(n-j)+\log p\,.$$

Помимо этого, на каждом j-м шаге над элементами матрицы производится порядка $2(n-1)(m+1)\cong 2nm$ операций. Число шагов алгоритма для системы равно n. Получаем временную сложность алгоритма:

$$T(n,\,p)=\sum_{j=1}^{n}\left(2nm+(n-j)+\log p\right)\cong O\left(n\cdot(nm+\log p)\right).$$

Заключение

Приведем сравнительный анализ асимптотической временной сложности предложенного алгоритма и алгоритмов, описанных в современной литературе, для системы n уравнений с m неизвестными в кольце вычетов $\mathbb{Z}_p$ ($p = \prod_{k=1}^{t} q_k^{\alpha_k}$) (см. табл.1).

Алгоритм	Временная сложность
Модифицированный метод Жордана	$O\left(n \cdot (nm + \log p)\right)$
Метод сведения к полям Галуа[1]	$O\left(n \cdot (n \cdot m \cdot \sum_{k=1}^{t} \alpha_k + \log p) + \sqrt{\ln p \ln\ln p} \cdot e^{\sqrt{\ln p \ln\ln p}}\right)$
Метод сведения к диофантовым уравнениям (с построением матрицы Смита)	$O\left(n^2 m^2 \log p\right)$

Табл.1.

Для проведения апостериорной оценки эффективности разработанного алгоритма в сравнении с аналогом, основанным на сведении исходной системы над кольцом вычетов к семейству систем над полями, который применялся на практике до появления настоящей работы, была разработана программа, реализующая вышеназванный алгоритм. Для решения систем в простых полях использовался алгоритм Жордана, для

[1] Оценка временной сложности этого метода дана при условии использования для разложения на множители числа p наиболее эффективного на сегодняшний день (см. [26]) алгоритма «квадратичного решета» Померанца, имеющего временную сложность $L(p)^{1+o(1)}$, где $L(p) = e^{\sqrt{\ln p \ln\ln p}}$

восстановления решения – Китайская теорема об остатках. Факторизация числа q осуществлялась с использованием алгоритма Миллера-Раббина (см. [26]) для генерации простых чисел a_i, не превосходящих $\sqrt{q}$, для каждого из которых затем осуществлялась проверка, не является ли a_i делителем q. Ввиду того, что в процессе испытаний использовались числа невысокого порядка, использование субэкспоненциального алгоритма разложения на множители представляется нецелесообразным (что подтвердили опытные данные[2], см. таблицу 2).

Испытания проводились с целью сравнить быстродействие программ на одних и тех же входных данных. В качестве входных данных выступали системы n уравнений с n неизвестными ($n = 500 \div 4000$) в кольце вычетов $\mathbb{Z}_q$ ($q = 31 \div 414141313$, простое или составное).

n - размерность системы	***q*** - порядок кольца	Время решения (в сек)	Алгоритм	Комментарий
500	31	2	СТАНД	Простое поле
500	31	3	МОДИФ	Простое поле
1000	22951	**25,375**	МОДИФ	Кольцо (22951=389x59)
1000	22951	**13,938**	МОДИФ ОПТ	Кольцо (22951=389x59)

[2] Испытания проводились на компьютере со следующими аппаратными характеристиками: процессор Intel Pentium IV 3,20GHz, ОЗУ 1Гб

Сокращения, использованные в таблице 2:

- СТАНД – алгоритм Жордана для решения систем в простых полях;
- РЕДУКЦИЯ – алгоритм, основанном на сведении исходной системы над кольцом вычетов к семейству систем над полями;
- МОДИФ - модификация алгоритма Жордана для решения систем в кольцах вычетов;
- МОДИФ ОПТ - оптимизированная модификация алгоритма Жордана для решения систем в кольцах вычетов.

1000	389	14,563	СТАНД	Простое поле
1000	59	15,459	СТАНД	Простое поле
Время на факторизацию:		0		
	Сумма:	**30.022**	РЕДУКЦИЯ	
1000	389	25,515	МОДИФ	Простое поле
1000	389	8,454	МОДИФ ОПТ	Простое поле
1000	59	26,219	МОДИФ	Простое поле
1000	59	8,843	МОДИФ ОПТ	Простое поле
2000	22331	**198.609**	МОДИФ	Кольцо (137x163)
2000	22331	**146.922**	МОДИФ ОПТ	Кольцо (137x163)
2000	137	118,203	СТАНД	Простое поле
2000	163	117,656	СТАНД	Простое поле
Время на факторизацию:		0		
	Сумма:	**235.859**	РЕДУКЦИЯ	
2000	137	150.391	МОДИФ ОПТ	Простое поле
2000	137	214.859	МОДИФ	Простое поле
2000	163	148.797	МОДИФ ОПТ	Простое поле
2000	163	201.578	МОДИФ	Простое поле
4000	414141313	**2142.297**	МОДИФ	Кольцо (99721x4153)
4000	99721	1202.672	СТАНД	Простое поле
4000	4153	924.782	СТАНД	Простое поле
Время на факторизацию:		0.032		
	Сумма:	**2159.454**	РЕДУКЦИЯ	

1000	282785411	**33.75**	МОДИФ	Кольцо (282785411=4337 x 65203)
1000	282785411	**28,156**	МОДИФ ОПТ	Кольцо (282785411=4337 x 65203)
1000	65203	14.313	СТАНД	Простое поле
1000	4337	16,891	СТАНД	Простое поле
Время на факторизацию:		0.016		
	Сумма:	**31,220**	РЕДУКЦИЯ	

Табл. 2.

На основании проведенных испытаний можно сделать следующие выводы:

1. Для систем большой размерности разработанный алгоритм работает значительно эффективнее аналога;
2. Априорные оценки зависимости асимптотической временной сложности алгоритма от параметров n, m, q подтверждаются опытными данными;
3. Оптимизация модифицированного алгоритма Жордана (сокращение вызовов процедуры, реализующей алгоритм Евклида) позволяет значительно снизить мультипликативную постоянную в оценке $O\left(n \cdot (nm + \log p)\right)$, хотя и не влияет на асимптотическую эффективность.

Принципиальное отличие разработанного алгоритма от существовавших ранее заключается в том, что для получения решения системы в кольце вычетов не требуется раскладывать число $p-1$ на множители; более того, не нужно проверять, является ли это число простым или составным, поскольку асимптотическая сложность алгоритма для этих случаев одинакова. Таким образом, предложенный метод столь же эффективен при решении систем линейных уравнений в кольцах вычетов, как и метод Жордана в полях Галуа.

В данной работе новыми являются следующие положения и результаты:

- Разработан алгоритм решения систем линейных уравнений в кольцах вычетов, по асимптотической сложности эквивалентный известным алгоритмам решении систем линейных уравнений в простых полях и не требующий факторизации числа $p-1$;
- Результаты проведенных исследований подтверждают, что разработанный алгоритм решения систем линейных уравнений в кольцах вычетов существенно снижает трудоемкость алгоритмов дискретного логарифмирования.
- Разработанный алгоритм реализован в программе, зарегистрированной Федеральной службой по интеллектуальной собственности, патентам и товарным знакам (Роспатент) [19] и Отраслевым Фондом Алгоритмов и Программ (ОФАП) [18].
- Программный продукт, реализующий разработанный алгоритм, является инструментальным средством криптоанализа, позволяющим более эффективно подходить к оценке надежности асимметричных криптографических алгоритмов.

Полученные теоретические результаты прошли апробацию на конференции «РусКрипто2006» (секция «Теория и практика создания систем

информационной безопасности»), XXXI и XXXII Международной молодежной научной конференции «Гагаринские чтения» (секция «Информационные и телекоммуникационные технологии») [34, 33], научно-технической конференции студентов, аспирантов и молодых специалистов «Информационные технологии в бизнесе» (секция «Разработка математического и программного обеспечения информационных систем»), Межвузовской конференции «Актуальные проблемы современных компьютеров» [35], Юбилейной студенческой научной конференции, посвященной 70-летию МГУПИ, XXXIII Международной конференции IT + S&E`06 «Информационные технологии в науке, образовании, телекоммуникации и бизнесе» [17], Международной студенческой школе-семинаре "Новые информационные технологии" [32], на Всероссийском конкурсе инновационных проектов аспирантов и студентов по приоритетному направлению "Информационно-телекоммуникационные системы" [31] и Федеральной школе-конференции инновационных проектов аспирантов и студентов, проводимой Научным парком МГУ в рамках программы «СТАРТ».

Современная криптография — это соревнование методов шифрования и криптоанализа. Каждый новый метод криптоанализа приводит к пересмотру безопасности шифров, к которым он применим. Результаты проведенных исследований показывают, что общая проблема логарифмирования в конечных полях не может считаться достаточно прочным фундаментом для построения криптографических систем.

Список литературы

1. Adleman L. A Subexponential Algorithm for the Discrete Logarithm with Application to Cryptography // Proceedings of the IEEE 20th Annual Symposium on Foundations of Computer Science (FOCS), 1979. P. 55 – 60.
2. Coppersmith D., Odlyzko A., Schroeppel R. Discrete logarithms in GF(p) // Algorithmica. 1986. V. 1. P. 1—15.
3. ElGamal T. A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms // IEEE Transactions on Information Theory, v. IT-31, n. 4, 1985. P. 469-472.
4. FIPS PUB 186. Digital Signature Standard (DSS).
5. FIPS PUB 186-2. Digital Signature Standard (DSS).
6. Frey G., Ruck H.-G. A Remark Concerning m-Divisibility and Discrete Logarithm in the Divisor Class Group of Curves // In Mathematics of Computation, 62, 1994. P. 865-874.
7. Gordon L.A., Loeb M.P., Lucyshyn W., Richardson R. CSI/FBI Computer Crime and Security Survey 2006. Computer Security Institute Publications, 2006.
8. Lukawiecki R. A-to-Z of Data Protection on the Windows Platform // Microsoft Tech-Ed IT Forum, Microsoft Corporation & Project Botticelli Ltd, 2006.
9. Odlyzko A.M. Discrete logarithms: The past and the future. AT&T Labs–Research, 1999.
10. Rivest R.L., Shamir A., Adleman L.M. A Method for Obtaining Digital Signatures and Public Key Cryptosystems// Communications of the ACM, v. 21, n. 2, Feb 1978. P. 120-126.
11. Schirokauer O. Discrete logarithms and local units. Phil. Trans. R. Soc. Lond. A., V. 345, 1993. P. 409—423.
12. Schneier B. Snake Oil, Crypto-Gram // February, 1999. Available via http://www.counterpane.com/Crypto-Gram.html

13. Western A.E., Miller J.C.P. Tables of indices and primitive roots. Cambridge University Press, 1968.
14. Авдошин С.М., Савельева А.А. Алгоритм решения систем линейных уравнений в кольцах вычетов // Информационные технологии. 2006. № 2. С.50-54.
15. Авдошин С.М., Савельева А.А. Криптоанализ: современное состояние и перспективы развития //Новые технологии; М.: Машиностроение, 2007. - 24 с. - (Библиотечка журнала "Информационные технологии"; Приложение к журналу "Информационные технологии";N 3).
16. Авдошин С.М., Савельева А.А. Криптографические методы защиты информационных систем // Известия АИН им. А.М. Прохорова. Бизнес-информатика. 2006. Т. 17. 92 С. 91-99.
17. Авдошин С.М., Савельева А.А. Методы повышения эффективности алгоритмов дискретного логарифмирования, использующих факторную базу // Труды XXXIII международной конференции «Информационные технологии в науке, образовании, телекоммуникации и бизнесе, IT + S&E`06», майская сессия, Украина, Крым, Ялта-Гурзуф, 20 - 30 мая, 2006. С. 133 – 134.
18. Авдошин С.М., Савельева А.А. Свидетельство об отраслевой регистрации разработки № 5410: «Программа решения систем линейных уравнений в кольцах вычетов». Зарегистрировано Государственным координационным центром информационных технологий в Отраслевом фонде алгоритмов и программ 23.11.05.
19. Авдошин С.М., Савельева А.А. Свидетельство об официальной регистрации программы для ЭВМ № 2005612258: «Программа решения систем линейных уравнений в кольцах вычетов». Зарегистрировано в Реестре программ для ЭВМ 02.09.2005.
20. Василенко О.Н. Теоретико-числовые алгоритмы в криптографии. М.: МЦНМО, 2004.

21. Глухов М.М., Елизаров В.П., Нечаев А.А. Алгебра: Учебник. В 2-х т. Т. I - М.: Гелиос АРВ, 2003.
22. ГОСТ Р34.10-01. Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи.
23. ГОСТ Р34.10-94. Информационная технология. Криптографическая защита информации. Процедуры выработки и проверки электронно-цифровой подписи на базе асимметричного криптографического алгоритма.
24. Джекобсон Н. Теория колец (Перевод с английского Н. Я. Виленкина). М.: Государственное издательство иностранной литературы, 1947.
25. Иванов М.А. Криптографические методы защиты информации в компьютерных системах и сетях. М.: КУДИЦ-ОБРАЗ, 2001.
26. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. М.: МЦНМО, 1999.
27. Кузнецов М.И., Бурланков Д.Е., Чирков А.Ю., Яковлев В.А. Компьютерная алгебра: Учебник. // Нижегородский Государственный Университет им. Н.И. Лобачевского, 2002. опубликовано: http://www.itlab.unn.ru/archive/docs/coaBook.pdf.
28. Ноден П., Китте К. Алгебраическая алгоритмика. Пер. с франц. - М.: Мир, 1999.
29. Ростовцев А.Г., Маховенко Е.Б. Теоретическая криптография. СПб.: АНО НПО Профессионал, 2005.
30. Ростовцев А.Г., Михайлова Н.В. Методы криптоанализа классических шифров // Опубликовано: http://crypto.hotbox.ru/download/cryptoan.zip, 1998.
31. Савельева А.А. Инструментальные средства криптоанализа // Сборник материалов Всероссийского конкурса инновационных проектов аспирантов и студентов по приоритетному направлению развития науки и

техники «Информационно-телекоммуникационные системы». М.: ГНИИ ИТТ «Информика», 2005. С. 44.

32. Савельева А.А. Исследование алгоритмов дискретного логарифмирования и способы повышения их эффективности // «Новые информационные технологии». Тезисы докладов XIV Международной студенческой школы-семинара" - М.: МИЭМ, 2006. С. 411-412.
33. Савельева А.А. Криптоанализ шифров, основанных на сложности задачи дискретного логарифмирования в конечных полях // XXXII Гагаринские чтения. Тезисы докладов Международной молодежной научной конференции. Т.4. М.: МАТИ, 2006. С. 34-35.
34. Савельева А.А. Метод решения систем линейных уравнений в кольцах вычетов // XXXI Гагаринские чтения. Тезисы докладов Международной молодежной научной конференции. Т.4. М.: МАТИ, 2005. С. 29-30.
35. Савельева А.А. Новый подход к решению систем уравнений в задачах дискретного логарифмирования // Программное и информационное обеспечение систем различного назначения на базе персональных ЭВМ: Межвузовский сборник научных трудов / Под ред. д. т. н., проф. Михайлова Б. М. М.: МГУПИ, 2006. Вып. 9. С. 193-197.
36. Семаев И. А. О сложности вычисления логарифмов на эллиптических кривых // Вторая международная конференция по теории чисел и ее приложениям, Тула, 1993.