

Подпись и шифрование на эллиптической кривой: анализ безопасности и безопасная реализация

Рассматривается безопасность стандартов электронной цифровой подписи ГОСТ Р 34.10–2001, DSS и алгоритма шифрования с открытым ключом на эллиптических кривых. Показано, что требования, предъявляемые к генератору случайных чисел при формировании подписи, строго выше, чем требования к генератору секретных ключей подписи. Стойкость протокола шифрования с открытым ключом не ниже, чем стойкость протокола электронной цифровой подписи. Определяется сложностная неоднородность криптосистемы как число задач, решение каждой из которых ведет к нарушению безопасности. Сформулирован принцип максимальной сложностной однородности, согласно которому стойкость криптосистемы нельзя повысить, увеличивая ее сложностную неоднородность. Предложены пути уменьшения сложностной неоднородности протоколов подписи и шифрования на основе усиления стандартных криптоалгоритмов.

Rostovtsev A. G., Makhovenko E. B. (SPbSPU)

Elliptic curve signcryption: analysis of security and secure implementation

The security of digital signature standards GOST R 34.10–2001, ECDSS and of the public-key encryption protocol, based on elliptic curves, is considered. It is shown that requirements to random number generator are strictly harder than requirements to secret key generator. Strength of the public key encryption protocol is not less than strength of the digital signature protocol. Cryptosystem complexity nonuniformity is determined as number of such problems, that solving any of them breaks the cryptosystem. Principle of maximum complexity uniformity is stated, according which strength of a cryptosystem cannot be increased by incrementing its complexity nonuniformity. Complexity uniformity can be incremented by using presented methods, based on strengthen of standard cryptographic algorithms.

1. Задача дискретного логарифмирования на эллиптической кривой

В основу безопасности российского стандарта электронной цифровой подписи ГОСТ Р 34.10–2001 на эллиптической кривой $E(\mathbf{F}_p)$ над простым конечным полем из p элементов и его американского аналога ECDSS положена задача дискретного логарифмирования на эллиптической кривой. Эллиптическая кривая $E(\mathbf{F}_p)$ в форме Вейерштрасса задается уравнением

$$y^2 \equiv f(x) \pmod{p}, \quad (1)$$

где кубический многочлен $f(x)$ не имеет кратных корней в поле \mathbf{F}_p .

Точки (x, y) кривой образуют конечную аддитивную абелеву группу, порядок $\#E(\mathbf{F}_p)$ которой согласно теореме Хассе близок к p [8]:

$$|\#E(\mathbf{F}_p) - p - 1| \leq 2\sqrt{p}.$$

По основной теореме об абелевых группах группа точек эллиптической кривой изоморфна прямой сумме циклических групп, порядки которых являются степенями простых чисел. Для эллиптических кривых этот результат можно уточнить: группа точек либо циклическая, либо является прямой суммой циклических групп, порядок одной из которых не больше квадратного корня из порядка второй. Если группа точек не циклическа, то число точек на кривой должно быть не свободным от кубов [4].

Задача дискретного логарифмирования формулируется следующим образом: для точки P , лежащей в циклической группе, образованной точкой Q , найти показатель l такой, что $P = lQ$. Для обеспечения максимальной безопасности задача дискретного логарифмирования должна быть максимально сложной. Отметим, что сложность дис-

кретного логарифмирования в значительной степени зависит от параметров задачи (уравнения эллиптической кривой, порядка циклической группы и др.).

Для повышения сложности задачи дискретного логарифмирования порядок r циклической группы должен быть большим простым числом. В противном случае можно найти «проекции» логарифма в циклических подгруппах простых порядков и восстановить логарифм по этим «проециям» по китайской теореме об остатках. Поэтому число точек на эллиптической кривой должно иметь большой простой делитель. Согласно ГОСТ Р 34.10–2001 число точек должно быть простым или кратным 2 (3, 4, ..., 7) простым числом, то есть число r должно быть короче числа p не более чем на 2 бита. Отсюда следует, что группа точек эллиптической кривой, допускаемой указанным стандартом, всегда циклическа.

Алгоритмы дискретного логарифмирования можно разделить на универсальные, применимые к произвольным эллиптическим кривым, и специальные, применимые только к эллиптическим кривым, обладающим некоторыми специфическими свойствами.

В настоящее время наилучшими универсальными алгоритмами дискретного логарифмирования на эллиптической кривой являются алгоритм Полларда [6] и алгоритм встречи на случайном дереве [1]. Алгоритм Полларда обладает временной сложностью $O(\sqrt{r})$, емкостной сложностью $O(\log r)$ и не допускает распараллеливания. Алгоритм встречи на случайном дереве обладает временной и емкостной сложностью $O(\sqrt{r \log r})$, но зато допускает распараллеливание для произвольного числа параллельно работающих процессоров.

Специальные алгоритмы дискретного логарифмирования используют инъективные гомоморфизмы из группы точек эллиптической кривой в аддитивную группу поля \mathbf{F}_p или в мультипликативную группу $\mathbf{F}_{p^k}^*$ расширенного поля. Из определения гомоморфизма и теоремы Хассе следует, что для существования гомоморфного вложения $E(\mathbf{F}_p) \rightarrow \mathbf{F}_p$ (являющегося изоморфизмом групп и вычислимого со сложностью $O(\log^3 p)$ методом логарифмической производной [7]) необходимо, чтобы выполнялось равенство $\#E(\mathbf{F}_p) = p$. Инъективный гомоморфизм $E(\mathbf{F}_p) \rightarrow \mathbf{F}_{p^k}^*$ задается спариванием Вейля [8] и существует, если $\#E(\mathbf{F}_p) \neq p$. Сложность вычисления такого гомоморфизма — полиномиальная от p^k [7]. Благодаря такому гомоморфизму можно свести задачу дискретного логарифмирования в группе точек эллиптической кривой к задаче дискретного логарифмирования в подгруппе порядка r группы $\mathbf{F}_{p^k}^*$ и, следовательно, к задаче дискретного логарифмирования во всей группе $\mathbf{F}_{p^k}^*$. В настоящее время наилучшим известным алгоритмом логарифмирования в группе $\mathbf{F}_{p^k}^*$ является метод решета числового поля (для $k = 1$) и метод решета поля функций (для $k > 1$). На сегодняшний день сложность этих алгоритмов оценивается как

$$O\left(\exp(1,53 \cdot \sqrt[3]{\ln p^k (\ln \ln p^k)^2})\right).^1$$

¹ Возможно, эту оценку можно улучшить, если воспользоваться несколькими последовательными квадратичными расширениями поля \mathbf{Q} (метод решета числового поля) или поля $\mathbf{F}_p(t)$ (метод решета поля функций).

Отсюда следует, что для обеспечения высокой сложности задачи дискретного логарифмирования на эллиптической кривой сложность задачи дискретного логарифмирования в группе $\mathbf{F}_{p^k}^*$ должна быть не менее $O(\sqrt{r})$.

Отметим два важных момента. Во-первых, сложность задачи дискретного логарифмирования на эллиптической кривой практически не снижается в течение 15 лет с момента опубликования первой криптосистемы на эллиптических кривых, чего нельзя сказать о сложности задачи дискретного логарифмирования в мультипликативной группе конечного поля. Во-вторых, логарифмирование на эллиптической кривой методом Полларда или встречи на случайном лесе не допускает предвычислений, уменьшающих сложность задачи, если хотя бы одна из точек P и Q не известна. Поэтому время эксплуатации информационной системы может быть продлено заменой персональных секретных и соответствующих открытых ключей. Логарифмирование в конечном поле $\mathbf{F}_{p^k}^*$ допускает предвычисления, не зависящие от точек P и Q , то есть замена персонального ключа практически не может продлить срок безопасной эксплуатации системы [1]. Поэтому сложность логарифмирования в группе $\mathbf{F}_{p^k}^*$ определяет не время действия персонального ключа подписи (или подписанного сообщения), а время жизни информационной системы в целом.

Если в правой части уравнения (1) многочлен задан неполным уравнением $f(x) = x^3 + Ax$ или $f(x) = x^3 + B$, то сложность логарифмирования может быть несколько снижена по сравнению с традиционной оценкой. Это обусловлено тем, что при переходе к алгебраическому замыканию поля \mathbf{F}_p эллиптическая кривая обладает автоморфизмами

соответственно вида $(x, y) \rightarrow (-x, \sqrt{-1}y)$ или $(x, y) \rightarrow \left(\frac{-1 + \sqrt{-3}}{2}x, -y \right)$. Эти

автоморфизмы образуют мультипликативную группу порядка 4 в первом случае и порядка 6 во втором случае. В результате появляется возможность разбить задачу дискретного логарифмирования на две последовательные подзадачи: сначала вычислить логарифм для орбиты точек P, Q относительно группы автоморфизмов, а затем уточнить логарифм внутри орбиты.

Таким образом, для обеспечения высокой сложности задачи дискретного логарифмирования на эллиптической кривой необходимо выполнение следующих условий, предусмотренных ГОСТ Р 34.10–2001:

- 1) p — простое число длины 256 бит;
- 2) многочлен $f(x)$ в правой части уравнения (1) не имеет кратных корней (дискриминант многочлена f отличен от нуля);
- 3) многочлен $f(x)$ не задается неполным уравнением;
- 4) число точек $\#E(\mathbf{F}_p)$ имеет простой делитель r длины от 254 до 256 бит;
- 5) $\#E(\mathbf{F}_p) \neq p$;
- 6) $p^k \neq 1 \pmod{r}$ для $k = 1, \dots, 30$.

Стандарт ECDSS не предусматривает ограничение по п. 3, число r должно иметь длину 160 бит.

Задача дискретного логарифмирования на эллиптической кривой сводится к задаче поднятия точки кривой из конечного поля в числовое поле [1].²

² Поднятие заключается в таком «доставании» координат точки эллиптической кривой положительными и отрицательными степенями числа p , что сравнение (1) переходит в равенство над числовым полем.

2. Безопасность протокола электронной цифровой подписи в целом

В основу протокола электронной цифровой подписи ГОСТ Р 34.10–2001 положен протокол Эль-Гамала [3]. Этот протокол обладает вычислимыми морфизмами, позволяющими на основании одного подписанного сообщения создавать произвольное число формально правильных пар сообщение–подпись, поэтому подпись вычисляется для значения хэш-функции от сообщения.

Пусть m — подписываемое сообщение, h — хэш-функция по ГОСТ Р 34.11–94, $\{E(\mathbf{F}_p), Q, P\}$ — открытый ключ, число l , — секретный ключ, при этом $P = lQ$. Для формирования подписи выполняются следующие действия:

- 1) вычисляется хэш-функция $e \equiv h(m) \pmod{r}$, причем $e \neq 0$;
- 2) вырабатывается случайный показатель k , $0 < k < r$, и вычисляется точка $R = (x_R, y_R) = kQ$, причем $x_R \neq 0 \pmod{r}$;
- 3) вычисляется часть подписи ³

$$s \equiv (lx_R + ke) \pmod{r}, \quad (2)$$

причем $s \neq 0$.

Подписанное сообщение представляет собой тройку $(m, x_R \pmod{r}, s)$.

Для проверки подписи выполняются следующие действия:

- 1) проверяются условия $0 < x_R \pmod{r} < r$ и $0 < s < r$;
- 2) вычисляется $e \equiv h(m) \pmod{r}$ и если $e = 0$, то считается $e = 1$;
- 3) вычисляется точка $R' = (se^{-1} \pmod{r})Q - (x_R e^{-1} \pmod{r})Q$;
- 4) проверяется сравнение $x_{R'} \equiv x_R \pmod{r}$. Если сравнение выполняется, то подпись верна.

Безопасность протокола подписи непосредственно зависит от сложности обращения хэш-функции и сложности вычисления коллизий хэш-функции, так как в первом случае можно вычислить сообщение для имеющейся подписи, а во втором — заготовить пару сообщений с одинаковыми значениями e , подписать одно из них, а потом заменить одно сообщение другим. Сложность вычисления коллизий хэш-функции методом Полларда равна $S_e = O(\sqrt{r})$. Однако сложность вычисления коллизий шаговой функции хэширования, при помощи которой вычисляется h , равна $S_e^{3/4}$ [2].

Этот протокол предполагает жесткие требования к генератору случайных чисел.

Теорема 1. Если случайное число k хоть однажды будет предсказуемым (вычислимым) для нарушителя или повторится в течение срока действия открытого ключа, то ключ создания подписи может быть вскрыт с полиномиальной сложностью.

Доказательство. Если число k известно нарушителю, то секретный ключ вскрывается решением уравнения (2) над полем \mathbf{F}_r : $l \equiv (s - ke)x_R^{-1} \pmod{r}$. Если в течение срока действия ключа число k повторится для сообщений m_1 и m_2 , по соответствующим подписанным сообщениям $(m_1, x_R \pmod{r}, s_1)$ и $(m_2, x_R \pmod{r}, s_2)$ ключ l можно вскрыть, решив систему двух линейных над \mathbf{F}_r уравнений $s_1 \equiv (lx_R + ke_1) \pmod{r}$, $s_2 \equiv (lx_R + ke_2) \pmod{r}$ относительно неизвестных k и l . ■

Отметим, что для вскрытия секретного ключа подписи достаточно не только предсказать случайное число, которое появится в будущем, но и найти случайное число, которое использовалось ранее. Например, если нарушитель получил доступ к алго-

³ В стандарте ECDSS m и s переставлены местами.

ритмическому генератору случайных чисел,⁴ допускающему восстановление предшествующих состояний из текущего состояния, то все электронные цифровые подписи, созданные на действующем ключе, должны считаться недействительными! Поэтому если к компьютеру, реализующему формирование подписи, возможен несанкционированный доступ (например, при отправке компьютера в ремонт в результате поломки) и генератор случайных чисел реализован алгоритмически, то алгоритм генерации случайных чисел должен быть вычислительно необратимым.

Для нахождения секретного ключа l по открытому ключу достаточно решить одну из задач дискретного логарифмирования — найти сам показатель l по точкам Q, P или один из показателей k по точкам Q и R (x_R легко восстанавливается по $x_R \pmod{r}$), а y_R можно вычислить как $\sqrt{f(x_R)} \pmod{p}$). Таким образом, требования, предъявляемые к генератору случайных чисел, оказываются более жесткими, чем к генератору ключей: нарушителю все равно, что искать: k или l , но повтор числа k в течение срока действия ключа приводит к вскрытию ключа l .⁵ Таким образом, справедливо следующее утверждение.

Теорема 2. Энтропия⁶ ключа создания подписи не превышает энтропии случайных чисел, вырабатываемых генератором.⁷

Примером неудачной реализации генератора случайных чисел является линейный конгруэнтный генератор. Пусть генератор вырабатывает числа вида $k_{i+1} \equiv sk_i + t \pmod{r}$, причем параметры s, t , а также два последовательных подписанных сообщения $(m_1, s_1, x_1 \pmod{r})$ и $(m_2, s_2, x_2 \pmod{r})$ известны нарушителю. Соответствующие уравнения создания подписи имеют вид: $s_1 \equiv lx_1 + ke_1 \pmod{r}$, $s_2 \equiv lx_2 + (sk + t)e_2 \pmod{r}$. В этих уравнениях все параметры, кроме k и l , известны нарушителю. Поскольку уравнения линейны относительно указанных неизвестных, ключ может быть вскрыт решением системы двух линейных уравнений.

Аналогично, если генератор реализует рекуррентное соотношение

$$k_{i+1} \equiv g(k_i) \pmod{r},$$

где g — алгебраическая функция, то задача вскрытия ключа сводится к решению системы алгебраических уравнений над полем \mathbf{F}_r .⁸

Назовем *сложностной неоднородностью криптосистемы* число математических задач таких, что решение любой из них ведет к нарушению безопасности. Если криптосистема не является безусловно стойкой, то сложностная неоднородность задается натуральным числом.

⁴ Такие генераторы называют также генераторами псевдослучайных чисел. Согласно определению Колмогорова, генератор вырабатывает случайную последовательность, если ее энтропия достаточно высока. Поскольку энтропия псевдослучайной последовательности может быть гарантированно большой, то последовательность можно называть случайной.

⁵ Отсюда следует, что если пользователю системы, в которой применяется электронная цифровая подпись, разрешается генерировать случайные числа k в ходе создания подписи, то запрет самому генерировать секретные ключи l создания подписи является бессмысленным.

⁶ Здесь и далее имеется в виду колмогоровская энтропия.

⁷ Отсюда вытекает потенциальная уязвимость устройств, реализующих электронную цифровую подпись, по отношению к атакам, провоцирующим различного рода сбои компьютера, имеющие целью нарушить безопасную работу генератора случайных чисел.

⁸ Такие уравнения решаются легко по сравнению с аналогичными уравнениями над числовыми полями или над полем \mathbf{C} комплексных чисел.

Теорема 3. Предположим, есть две однотипных криптосистемы, нарушение безопасности которых сводится к решению одной из двух математических задач. В первой криптосистеме эти задачи одинаковы (A и A), а во второй различны (A и B). Тогда стойкость второй криптосистемы не может превысить стойкость первой.

Доказательство. Отображение множества математических задач в множество их сложностей задает линейную упорядоченность задач по сложности (при этом сложность задачи может меняться во времени). С таким отношением порядка множество задач образует дистрибутивную решетку. Сравним сложности S_1 и S_2 взлома первой и второй криптосистем. Обозначим через S_A, S_B сложности задач A и B соответственно. Для первой криптосистемы $S_1 = \min(S_A, S_A) = S_A$, для второй криптосистемы $S_2 = \min(S_A, S_B)$. По свойству поглощения для решеток выполняется условие $S_A \geq \min(S_A, S_B)$, откуда $S_1 \geq S_2$. ■

Следствие 4. Введением в криптосистему наряду с существующим списком задач дополнительной математической задачи невозможно повысить стойкость криптосистемы.

Доказательство выполняется индукцией по множеству задач.

Чем меньше сложностная неоднородность криптосистемы, тем потенциально более стойкой является эта криптосистема (добавление новой задачи не может повысить стойкость, а может ее только снизить). Поэтому при проектировании криптосистем следует соблюдать *принцип максимальной сложностной однородности*: **число задач, положенных в основу безопасности криптосистемы, таких, что решение любой из них нарушает безопасность, должно быть минимальным.**⁹ Оптимальной является криптосистема со сложностной неоднородностью 1 (ее безопасность основана на единственной задаче).

Безопасность электронной цифровой подписи базируется на следующих предположениях:

- 1) задача дискретного логарифмирования на эллиптической кривой является сложной (решение этой задачи ведет к вскрытию ключа);
- 2) энтропия генератора случайных чисел не ниже энтропии генератора ключей (в противном случае стойкость подписи будет снижена по сравнению с расчетной оценкой);
- 3) вероятность повтора двух случайных чисел в течение срока действия ключа пренебрежимо мала (в противном случае можно рассчитывать на вскрытие секретного ключа с низкой сложностью);
- 4) хэш-функция является вычислительно необратимой (в противном случае можно подменить подписанное сообщение другим);
- 5) коллизии хэш-функции являются трудновычислимыми (в противном случае можно заготовить пару сообщений, составляющих коллизию, и после подписи заменить подписанное сообщение другим).

Эти предположения определяют список задач, положенных в основу безопасности, неоднородность стандартного протокола подписи равна 5. Можно считать, что стойкость подписи не превышает минимума сложностей дискретного логарифмирования, нахождения используемого случайного числа, обращения хэш-функции, вычисления коллизий хэш-функции.

⁹ Очевидный практический выигрыш от использования принципа максимальной однородности — минимизация трудовых затрат при анализе безопасности криптосистемы как на этапе разработки, так и на последующих этапах периодической переаттестации..

Протокол подписи может быть легко усилен [1], причем это усиление является строгим, то есть не вносит слабостей по сравнению со стандартом при прочих равных условиях. Усиление заключается в сцеплении сообщения m и координаты $x_R \pmod{r}$ случайной точки R через хэш-функцию: вместо $e \equiv h(m) \pmod{r}$ следует использовать $e \equiv h(m || x_R \pmod{r}) \pmod{r}$. Это исключит уязвимость протокола, связанную с заготовкой коллизий (задача п. 5) и вызванную этим необходимость периодической смены стартового вектора хэш-функции. Такое усиление протокола подписи снизит его неоднородность до 4, при этом задача обращения хэш-функции заменяется задачей ее *ограниченного обращения*: часть разрядов аргумента, определяемая точкой R , должна иметь заданный вид. Очевидно, что задача ограниченного обращения не может быть проще задачи обращения (противоположное предположение немедленно ведет к противоречию).

3. Протокол шифрования с открытым ключом и его безопасность

В ряде случаев, например, для доставки ключей симметричного шифрования, оказывается удобным использование шифрования с открытым ключом. Такие протоколы не предусмотрены действующим стандартом, поэтому авторы сочли возможным привести вариант протокола.

Общими для всех пользователей параметрами системы являются эллиптическая кривая $E(\mathbb{F}_p)$ и образующая точка Q . Персональным открытым ключом шифрования является точка P . Секретным ключом расшифрования является показатель l такой, что $P = lQ$.

Протокол 1 (шифрование с открытым ключом)

Вход отправителя: сообщение m , $0 < m < r$.

Выход отправителя: шифрограмма.

Вход получателя: показатель l такой, что $P = lQ$; шифрограмма.

Выход получателя: сообщение m .

Зашифрование (действия отправителя):

- 1) сгенерировать случайный показатель k , вычисляет точку $R = kQ$;
- 2) вычислить точку $S = kP = (x_S, y_S)$;
- 3) вычислить

$$c \equiv (x_S + m) \pmod{r}. \quad (3)$$

Зашифрованный текст представляет собой пару (R, c) .

Расшифрование (действия получателя):

- 1) вычислить точку $S = lR = (x_S, y_S)$;
- 2) найти открытый текст $m \equiv (c - x_S) \pmod{r}$.

Задача Диффи — Хеллмана формулируется следующим образом: для заданных точек $Q, P = lQ, R = kQ$ найти точку $klQ = kP = lR$. Эта задача полиномиально сводится к задаче дискретного логарифмирования на эллиптической кривой, так как для ее решения достаточно найти один из показателей k или l . Обратная сводимость задач, означающая их полиномиальную эквивалентность, в настоящее время установлена только для частного случая, когда число $r - 1$ имеет гарантированно малые простые делители [1]. Однако методы решения задачи Диффи — Хеллмана, отличные от дискретного логарифмирования, не известны, поэтому обычно полагают, что указанные задачи эквивалентны. Поэтому можно считать, что безопасность протокола 1 основана на задаче дискретного логарифмирования.

Рассмотренный протокол сохраняет некоторую информацию о сообщении m . Это обусловлено тем, что x_S не является полностью непредсказуемой величиной, а удовлетворяет условию для символа Якоби: $\left(\frac{f(x_S)}{p}\right) = 1$. Это равносильно знанию одного бита информации о секретной величине x_S и, следовательно, о сообщении m . Для того чтобы устранить этот недостаток желательно выбирать сообщение m короче чем r и разбавлять его случайными битами на фиксированных позициях. Поскольку энтропия сообщения при известной шифрограмме не превышает $\frac{1}{2} \log_2 r$ (это оценка для логарифма стойкости схемы шифрования), то данное требование не является обременительным. Поэтому для безопасной передачи ключа ГОСТ 28147–89, имеющего энтропию в случае анализа на основе подобранных открытых текстов не более 255 бит,¹⁰ нужно выбирать эллиптическую кривую с величиной $\log_2 r \geq 510$. Для сравнения: протоколы Диффи — Хеллмана или шифрования с открытым ключом, основанные на задаче дискретного логарифмирования в конечном поле, имеют энтропию $c\sqrt{\ln p(\ln \ln p)^2}$ для небольшой константы c . Ключевое соглашение не снизит стойкость шифра, если длина простого числа p будет составлять не менее 24 килобит.

Требования, предъявляемые к генератору случайных чисел в протоколе 1, менее жесткие, чем в протоколе подписи. Это обусловлено тем, что однократное вычисление случайного числа k ведет к расшифрованию сообщения m , зашифрованного с помощью этого числа, но не к вскрытию секретного ключа. Аналогично повторение случайного числа ведет к расшифрованию одного из сообщений, если другое сообщение, зашифрованное с помощью этого случайного числа, известно.

Сложностная неоднородность протокола 1 равна 3, причем его безопасность основана на тех же задачах, что и безопасность протокола подписи. Поэтому можно утверждать: если задачи дискретного логарифмирования и Диффи — Хеллмана эквивалентны, то протокол 1 не менее безопасен, чем протокол подписи ГОСТ Р 34.10–2001.

4. Пути снижения сложностной неоднородности

Реализация протокола электронной цифровой подписи ГОСТ Р 34.10–2001 требует использования генератора случайных чисел, к которому предъявляются весьма жесткие требования.

Предположим, что в информационной системе возможны только явные компрометации ключей и аппаратуры, то есть обо всех случаях несанкционированного доступа к ключам и информации, определяющей стойкость ключа, становится известно. Предположим, что секретный ключ создания подписи вводится извне. Однако для алгоритмического генератора случайных чисел это невозможно: все состояния генератора должны быть различны. Поэтому текущее состояние генератора должно храниться в компьютере и сохраняться в выключенном состоянии.

В этом случае алгоритмический генератор случайных чисел, обеспечивающий минимальную неоднородность, должен удовлетворять следующим требованиям:

- 1) задача вычисления случайного числа при известных параметрах генератора должна быть безусловно безопасна или эквивалентна задаче дискретного логарифмирования на эллиптической кривой;

¹⁰ Благодаря эндоморфизму ГОСТ 28147–89 как фейстелева шифра ключ можно искать с точностью до старшего разряда ключа [2].

- 2) вероятность повторения одинаковых случайных чисел в выборке объема $3 \cdot 10^9$ не должна превышать 10^{-9} .¹¹
- 3) задача вычисления предыдущего случайного числа по известному текущему состоянию генератора должна быть эквивалентна задаче дискретного логарифмирования на эллиптической кривой.

Указанным требованиям удовлетворяет следующий рекуррентный генератор, выполняющий вычисления на той же эллиптической кривой, которая используется в стандарте цифровой подписи. В качестве начального состояния может использоваться произвольная аффинная точка (x, y) эллиптической кривой. Пусть $R_i = (x_i, y_i)$ — текущая точка эллиптической кривой. Последующее состояние задается уравнением

$$R_{i+1} = (i + x_{R_i})Q. \quad (4)$$

В качестве выхода используется несколько (например, 16) младших битов точки x_{R_i} . Таким образом, для получения случайного числа необходимо последовательно пройти 16 состояний генератора. Зависимость текущего показателя в формуле (4) от номера состояния i позволяет избежать заикливания генератора и снижает вероятность повтора случайного числа. Текущее состояние генератора является секретным. Поскольку на каждом шаге текущее состояние меняется, то анализ сигналов, несущих информацию о секретном параметре генератора, методом накопления и выделения их из шума невозможен.

Если в результате явной компрометации текущее состояние R_{i+1} генератора стало известным, то восстановление предыдущих состояний эквивалентно дискретному логарифмированию: для точек Q и R_{i+1} нужно найти показатель $i + x_{R_i} \pmod{r}$ в уравнении (4). Поэтому третье требование к генератору выполнено.

Эксперимент на эллиптических кривых над малыми полями показал, что закон распределения текущих состояний генератора близок к равномерному. В силу того, что эллиптические кривые над малыми и большими простыми конечными полями описываются одинаковыми алгебраическими выражениями, можно предположить, что распределение состояний генератора близко к равномерному и в случае большого поля. Тогда второе требование к генератору выполняется с большим запасом.

Вычисление очередного случайного числа эквивалентно определению текущего состояния генератора, а состояния распределены примерно равномерно и начальное состояние безопасно. Предсказание такого генератора эквивалентно перебору начальных состояний, генератор в этом смысле является теоретически безопасным [5]. Поэтому и первое требование выполнено. Использование такого генератора позволяет снизить сложностную неоднородность протокола электронной цифровой подписи до 2.

Этот генератор может быть применен и для шифрования с открытым ключом, сложностная неоднородность указанного протокола составит 1, то есть задает оптимум в части сложностной неоднородности.

Может быть, задача обращения генератора, заданного рекуррентным уравнением $R_{i+1} = (i + x_{R_i})R_i$, была бы более сложной, чем задача обращения генератора (4), но эта задача отлична от задачи дискретного логарифмирования, и согласно следствию 4 такой генератор не может усилить криптосистему.

Для минимизации сложностной неоднородности протокола подписи следует заменить или дополнить «инородную» хэш-функцию ГОСТ Р 34.11–94 так, чтобы ее невозможно было обратить, не решив задачу дискретного логарифмирования. Поскольку

¹¹ Этот объем выборки соответствует генерации 10 подписей каждую секунду круглосуточно в течение 10 лет, а вероятность выбрана достаточно малой, чтобы ей можно было пренебречь.

проблема, связанная с коллизиями хэш-функции, может быть устранена указанным ранее усилением протокола, достаточно предложить хэш-функцию, ограниченное обращение которой требует решения задачи дискретного логарифмирования.

Предлагаемое усиление не отменяет стандартную хэш-функцию, а лишь дополняет ее применительно к протоколу подписи. Пусть m — аргумент стандартной хэш-функции $h(m)$. В уравнении (2) вместо $h_E(m)$ используется усиленная хэш-функция $h_E(m) \equiv x_{h(m)Q} \pmod{r}$. Поскольку $h_E(m)$ непосредственно зависит от $h(m)$, естественно предположить, что для обращения хэш-функции $h_E(m)$ необходимо предварительно найти $h(m)$. Это предположение представляется бесспорным, однако строго доказать его, по-видимому, очень непросто. Если задача обращения хэш-функции h окажется слабой, то вычисление промежуточного значения $h(m)$ (по предположению необходимое для нахождения m) по заданному значению $x_{h(m)Q} \pmod{r}$ эквивалентно дискретному логарифмированию. Таким образом, можно считать, что задача обращения усиленной хэш-функции $h_E(m)$ требует вычисления дискретного логарифма на эллиптической кривой. Эллиптическая кривая в хэш-функции должна быть та же, что и в схеме подписи.

5. Защита от атак на основе анализа опасных сигналов

Часто криптографические средства защиты информации используют долговременный секретный ключ, обращение к которому выполняется многократно при каждом зашифровании/расшифровании (выработке подписи). Процесс обработки информации сопровождается физическими полями (опасными сигналами), несущими информацию о ключе. Обычно уровень опасных сигналов значительно ниже уровня шума, однако использование методов оптимальной обработки сигналов (накопления) позволяет выделить их из шума и эти понизить стойкость шифра. Для этого необходимо, чтобы сигнал повторялся необходимое число раз в известные нарушителю моменты времени.

Защита от таких атак может быть осуществлена алгоритмически. При этом на каждой реализации процесс обработки информации случайным образом изменяется так, что затрудняет использование метода накопления. Рассмотрим способы такого противодействия применительно к протоколам подписи и шифрования с открытым ключом.

В случае электронной цифровой подписи защищаемыми параметрами являются ключ l и случайное число k . При этом генератор случайного числа представляется безопасным по отношению к лабораторным методам анализа, поскольку не использует одну и ту же секретную информацию для выработки очередного случайного числа.¹² По той же причине вычисление точки $R = kQ$ не представляется опасным.

Рассмотрим варианты защиты секретного ключа l подписи. Обращение к ключу l выполняется только в процессе вычисления подписи (2). Защиту от лабораторных методов можно реализовать двояко. С одной стороны, в (2) можно случайным образом изменять очередность умножений $lx_R \pmod{r}$ и $ke \pmod{r}$ и случайным образом менять порядок вызова операндов. С другой стороны, умножение больших чисел обычно реализуется алгоритмом «в столбик», при этом каждое слово одного сомножителя умножается на каждое слово другого. Поэтому порядок выбора слов и сомножителей тоже может быть случайным и не обязательно начинаться с младших разрядов. Это эквивалентно многократному уменьшению соотношения «сигнал/шум» при измерении пара-

¹² Генератор на основе симметричного шифра в режиме гаммирования этим качеством не обладает и поэтому вычисление случайного числа сводится к вычислению ключа генератора.

метров сигнала, так как нарушитель должен будет решить задачу, в какой момент времени обрабатывается данное слово ключа.

В случае шифрования с открытым ключом защищаемым параметром является секретный ключ l , используемый при вычислении точки lR в ходе расшифрования. Предлагается следующий алгоритм вычисления lR . Сначала вычисляются точки n_iR для $n_i = 2, 3, \dots, 15$. Этот этап не зависит от l . Показатель l представлен тетрадами $l = l_0 + 2^4l_1 + 2^8l_2 + \dots + 2^{252}l_{63}$. На втором этапе для тетрады l_{63} выбирается соответствующая точка $l_{63}R$ и четыре раза удваивается, затем к ней прибавляется точка $l_{62}R$, соответствующая тетраде l_{62} , сумма четыре раза удваивается и т. д.

Значение точки l_iR не содержит энтропии о тетраде l_i ключа, так что единственным опасным сигналом является сигнал, соответствующий номеру выбираемой точки в созданной на первом этапе базе данных. Для защиты сигналов, несущих информацию о ключе, можно начинать первый шаг этапа 2 случайным образом, не дожидаясь окончания этапа 1. Кроме того, из равенств $S = lR = -(r - l)R$ и $x_S = x_{-S}$ следует, что можно точку R умножать не на l , а на $(r - l)$. Случайное чередование показателей l и $r - l$ затруднит восстановление сигнала, так как при каждом измерении сигнала нарушителю придется решать, прямое или инверсное значение ключа используется.

Отмеченные недостатки и изложенные подходы справедливы и для американского стандарта подписи ECDSS на эллиптических кривых.

Библиографический список

1. **Ростовцев А. Г., Маховенко Е. Б.** Введение в криптографию с открытым ключом. — СПб.: Мир и Семья, Интерлайн, 2001.
2. **Ростовцев А. Г., Маховенко Е. Б.** Введение в теорию итерированных шифров. — СПб.: Мир и Семья, Интерлайн, 2002.
3. **ElGamal T.** A public key cryptosystem and a signature scheme based on discrete logarithms // IEEE Transactions on Information Theory, v. IT-31, 1985, pp. 469–472.
4. **Koblitz N.** A Course in Number Theory and Cryptography. — Springer-Verlag, 1987.
5. **Menezes A., van Oorschot P., Vanstone S.** Handbook of applied cryptography. — CRC Press, 1997.
6. **Pollard J.** Monte Carlo methods for index computation (mod p) // Mathematics of Computation, v. 32, 1978, pp. 918–924.
7. **Semaev I. A.** Evaluation of discrete logarithms in a group of p -torsion points of an elliptic curves in characteristic p // Mathematics of Computation. v. 67, n. 221, 1998, pp. 353–356.
8. **Silverman J. H.** The arithmetic of elliptic curves. — Springer-Verlag, 1986.