Федеральное государственное автономное образовательное учреждение высшего образования
Национальный исследовательский ядерный университет «МИФИ»

Кафедра «Криптология и кибербезопасность»

Криптографические протоколы

курс лекций

Запечников Сергей Владимирович,
профессор кафедры «Криптология
и кибербезопасность» НИЯУ МИФИ

Москва – 2018

Криптографические протоколы

курс лекций

Лекция 6.
Обеспечение стойкости криптосистем к компрометации ключей

12 марта 2018 г.

Пороговая криптография

Постановка задачи

Предположим, есть важная секретная информация, которую можно потерять. Ее опасно доверять кому-то одному. Возникает вопрос, как повысить надежность и безопасность ее хранения?

Первый путь **– сделать несколько копий этих данных и хранить их в разных местах. Резервирование обеспечивает высокую надежность хранения, но если скомпрометирована хотя бы одна копия, то секретность всей информация будет потеряна.**

Второй путь **– разделить секрет на несколько частей и хранить их в разных местах, при необходимости собирая вместе. Самый простой способ разделить секрет *s* на *n* частей – выбрать *n*-1 случайное число $s_1, s_2, ..., s_{n-1}$, а *n*-ю часть определить так: $s_n = s \oplus s_1 \oplus s_2 \oplus ... \oplus s_{n-1}$. Каждое число $s_i, i = \overline{1,n}$ носит название *доли секрета* (share). Доли создает носитель секрета *s*. Иногда это один из *n* участников, получающих доли, иногда – постороннее лицо, которое в этом случае называется *дилер*. Каждая доля s_i должна быть передана соответствующему участнику конфиденциально. Если эту долю видят и другие участники протокола, эта схема уже не работает. Для восстановления секрета *s* необходимо присутствие всех *n* сторон, имеющих доли секрета, которые должны выполнить операцию сложения: $s = s_1 \oplus s_2 \oplus ... \oplus s_n$. Так приходим к идее *схем разделения секрета* *(secret sharing scheme)* - сокращенно СРС. Такая схема обеспечивает высокую конфиденциальность (чтобы восстановить секрет, надо получить все его доли), но низкую надежность (если потеряна хотя бы одна доля, восстановить секрет уже будет невозможно).**

Пороговые схемы разделения секрета (СРС)

Мы хотим построить более гибкую схему. Пусть есть *n* участников криптосистемы. Мы хотим, чтобы любые *t* из них могли восстановить секрет, но никакие *t*-1 из них не смогли бы получить информацию о секрете. Число *t* (*t*<*n*) – параметр схемы, называемый порогом. Схема, обладающая такими свойствами, называется *(t,n)-пороговой СРС*. Она лучше, чем предыдущая, так как, если кто-то из участников потеряет свою долю или не будет участвовать в восстановлении секрета, секрет все равно можно восстановить и без этих долей.

Известны несколько математических методов реализации такой схемы. Однако далеко не все из них удобны на практике.

Геометрическая интерпретация пороговых СРС

Схема Шамира **(*A. Shamir, 1979*) основана на хорошо известном математическом факте, который заключается в том, что через любые *t* точек на плоскости можно провести бесконечное множество кривых, описываемых многочленом *t*-го порядка, но через любые *t*+1 различные точки можно провести только единственную кривую, описываемую многочленом *t*-го порядка. Так, через любую точку на плоскости проходит бесконечное множество прямых линий, но через две различные точки – только единственная. Через любые две точки можно провести бесконечное множество парабол, но через любые три различные точки – только одну и т.д. Таким образом, если каждому из участников криптосистемы «выдать» по одной точке, то восстановить кривую можно будет только при достаточном количестве участников.**

Математические основы СРС Шамира

В криптосистемах широко используется пороговая СРС Шамира, так как она допускает удобную геометрическую интерпретацию и легко обобщается для многочленов над конечными полями. Пусть *F* – конечное поле, $f(x)=a_0+a_1x+...a_{t-1}x^{t-1}, a_i \in F$ **– многочлен над полем *F*, т.е.** $f(x) \in F[x]$**. Известно, что такой многочлен обладает следующими свойствами.**

***1. Интерполируемость*. По данным *t* точкам многочлена:** $(x_1, y_1),...,(x_t, y_t)$**, где все** $x_1,...,x_t$ **различны,** $y_i = f(x_i)$**, можно найти его коэффициенты** $a_0, a_1,...,a_{t-1}$**. Алгоритм, делающий это, называется алгоритмом интерполяции.**

***2. Секретность*. По данным любым *t*-1 точкам полинома:** $(x_1, y_1),...,(x_{t-1}, y_{t-1})$**, где** $y_i = f(x_i)$**, никто не может ничего предполагать об a_0 – свободном члене *f(x)*.**

Эти свойства делают многочлены над конечными полями инструментом для построения пороговых криптосистем.

Протоколы СРС Шамира (1)

Рассмотрим (*t*,*n*)-пороговую СРС Шамира над полем Z_p, где *p* – большое простое число. Схема включает несколько протоколов.

Фаза инициализации. **Дилер *D* выбирает *n* различных ненулевых элементов поля Z_p, которые обозначаются $x_i, 1 \le i \le n, p \ge n+1$ – это точки, к которым «привязаны» участники. Часто выбирают $x_i \equiv i$, т.е. просто всем участникам схемы присваиваются порядковые номера. *D* передает x_i участнику P_i.**

***Распределение долей*:**

1) *D* хочет разделить секретный ключ $K \in Z_p$. *D* секретно, случайно и независимо друг от друга выбирает *t*-1 элемент поля $a_1,\ldots,a_{t-1}$, где $a_i \in Z_p$;

2) *D* конструирует многочлен степени, меньшей либо равной *t*-1, и вычисляет для $i = \overline{1,n}$: $y_i = a(x_i)$, где $a(x) = K + \sum_{m=1}^{t-1} a_m x^m \pmod p$, т.е. $K = a(0)$. Коэффициенты многочлена дилер хранит в секрете;

3) дилер по секретному и аутентичному каналу рассылает каждую из долей y_i соответствующему участнику P_i для всех $i = \overline{1,n}$.

Протоколы СРС Шамира (2)

Восстановление секрета **возможно двумя способами.**

I способ. Предположим, участники $P_{i_1},...,P_{i_t}$ **хотят восстановить секретный ключ** ***K*****. Они имеют** (x_{i_j}, y_{i_j}) **и знают, что** $y_{i_j} = a(x_{i_j}), j = \overline{1,t}$ **, где** $a(x) \in Z_p[x]$ **– неизвестный многочлен, выбранный** ***D*****. Так как** $a(x)$ **имеет степень, меньшую либо равную** ***t*****-1,** $a(x)$ **может быть записан в виде:**

$$a(x) = a_0 + a_1 x + ... + a_{t-1} x^{t-1},$$

где $a_0, a_1, ..., a_{t-1}$ **– неизвестные элементы поля** Z_p**, и** $a_0 = K$ **.**

Они получают систему ***t*** **линейных уравнений с** ***t*** **неизвестными** $a_0, a_1, ..., a_{t-1}$ **над полем** Z_p**:**

$$y_{i_j} = a_0 + \sum_{m=1}^{t-1} a_m x_{i_j}^m, j = \overline{1,t}.$$

Если уравнения линейно независимы, система имеет единственное решение. Они могут решить систему относительно неизвестных коэффициентов и получить a_0**.**

Утверждение. **(*t*,*n*)-пороговая СРС Шамира позволяет однозначно восстанавливать секрет любой группе из** ***t*** **участников схемы и обеспечивает совершенную секретность (теоретико-информационную стойкость) против попытки вычисления секрета любой группой из** ***j*** **участников, обладающих неограниченной вычислительной мощностью (*j*<*t*).**

Протоколы СРС Шамира (3)

<u>II способ.</u> Восстановить разделенный секрет можно и другим, более простым способом. Воспользуемся интерполяционной формулой Лагранжа:

$$a(x)=\sum_{j=1}^{t} y_{i_j} \prod_{\substack{1\le k\le t,\\ k\ne j}} \frac{x-x_{i_k}}{x_{i_j}-x_{i_k}}.$$

Мы имеем *t* пар чисел $(x_{i_j}, y_{i_j}), j=\overline{1,t}$ **, и доказали, что многочлен единственный. Следовательно, формула Лагранжа даст нам единственный верный результат.**

Формулу можно упростить, так как участникам группы не нужно вычислять все коэффициенты многочлена, а только свободный член *K=a(0)*:

***x*=0** $$K=\sum_{j=1}^{t} y_{i_j} \prod_{\substack{1\le k\le t,\\ k\ne j}} \frac{x_{i_k}}{x_{i_k}-x_{i_j}}.$$

Обозначим коэффициенты интерполяции в формуле Лагранжа: $b_j=\prod_{\substack{1\le k\le t,\\ k\ne j}} \frac{x_{i_k}}{x_{i_k}-x_{i_j}}, j=\overline{1,t}$ **. Они могут быть вычислены предварительно, так как все** x_{i_j}, x_{i_k} **общеизвестны. Остается вычислить ключ как линейную комбинацию *t* долей секрета:** $K=\sum_{j=1}^{t} b_j y_{i_j}$ **.**

Схема проверяемого разделения секрета Фельдмана (1)

В СРС Шамира нечестный дилер *D* может раздать участникам $P_1,...,P_n$ несовместные доли, из которых они никогда не восстановят секретный ключ *K*. Необходимо предложить такую схему, в которой можно было бы проверить совместимость долей секрета. Известны две СРС, решающих эту задачу, основанные на сложности задачи дискретного логарифмирования: СРС Фельдмана и СРС Педерсена.

Пусть *p,q* – большие простые числа, $p-1\equiv 0(\mathrm{mod}\, q)$. *g* – элемент порядка *q* группы Z_p^*, т.е. $g^q\equiv 1(\mathrm{mod}\, p)$. Для любой доли y_i вычисляется открытая величина $z_i=g^{y_i}(\mathrm{mod}\, p)$, которая по свойству гомоморфизма функции экспоненцирования позволяет каждому P_i проверять, что его собственная доля секрета совместима с открытой информацией.

D **выбирает многочлен $a(x)\in Z_q[x]$ с коэффициентами $a_0=K,a_1,...,a_{t-1}$ и раздает всем участникам соответствующие проверочные значения $g^{a_0},g^{a_1},..,g^{a_{t-1}}$.**

Положим $x_i\equiv i, i=\overline{1,n}$. Дилер *D* секретно передает каждому участнику схемы P_i предназначенную ему долю $y_i=a(i)\mathrm{mod}\, q$.

Схема проверяемого разделения секрета Фельдмана (2)

Каждый участник P_i проверяет свою долю, используя проверочное уравнение:

$$g^{y_i} \stackrel{?}{=} \left(g^{a_0}\right) \cdot \left(g^{a_1}\right)^i \cdot \left(g^{a_2}\right)^{i^2} \cdot \ldots \cdot \left(g^{a_{t-1}}\right)^{i^{t-1}} \left(\mathrm{mod}\, p\right).$$

В случае положительного результата проверки P_i распространяет всем остальным участникам схемы сообщение, что он принял свою долю, так как $y_i = a_0 + a_1 i + a_2 i^2 + \ldots + a_{t-1} i^{t-1} (\mathrm{mod} q)$. В случае отрицательного результата он делает вывод, что ему дилером была выдана неверная доля.

Если все $P_i, i = \overline{1,n}$ распространили сообщения о принятии долей, фаза распределения долей завершилась успешно. Такая же проверка может выполняться при восстановлении секрета.

Заметим, что в схеме проверяемого разделения секрета каждый может проверить только свою долю, но не чужую – для этого нужны схемы публично проверяемого разделения секрета.

Схема проверяемого разделения секрета Педерсена (1)

Числа *p,q,g,K* определяются так же, как и в предыдущей схеме. $h \in Z_p^*$ **– открытое общедоступное число, но такое, что** $d \in Z_q$**, где** $g^d = h(\mathrm{mod}\ p)$ **неизвестно.**

Чтобы распределить секрет *K*, дилер выбирает два многочлена $\delta(\cdot), \gamma(\cdot)$ **степени *t*-1 над полем Z_q с коэффициентом** $\delta_0 = K$ **и случайными коэффициентами** $\{\delta_m\}_{m\in\{1,\ldots,t-1\}}$ **и** $\{\gamma_m\}_{m\in\{0,\ldots,t-1\}}$ **соответственно, т.е.**

$$\delta(z) = \delta_0 + \delta_1 z + \delta_2 z^2 + \ldots + \delta_{t-1} z^{t-1} \in Z_q[z], \qquad \delta_0 = K,$$

$$\gamma(z) = \gamma_0 + \gamma_1 z + \gamma_2 z^2 + \ldots + \gamma_{t-1} z^{t-1} \in Z_q[z], \qquad \gamma_0 \text{ - случ.,}$$

и распространяет всем участникам схемы $P_i, i = \overline{1,n}$ **величину** $\varepsilon_m = g^{\delta_m} \cdot h^{\gamma_m} (\mathrm{mod}\ p), m = \overline{0,t-1}$**. Затем дилер *D* секретно пересылает всем P_i,** $i = \overline{1,n}$ **их доли** $\{u_i, w_i\}$**, где** $u_i = \delta(i), w_i = \gamma(i)$**.**

Проверочное уравнение для участника P_i:

$$g^{u_i} h^{w_i} \stackrel{?}{=} (\varepsilon_0) \cdot (\varepsilon_1)^i \cdot (\varepsilon_2)^{i^2} \cdot \ldots \cdot (\varepsilon_{t-1})^{i^{t-1}} (\mathrm{mod}\ p).$$

Схема проверяемого разделения секрета Педерсена (2)

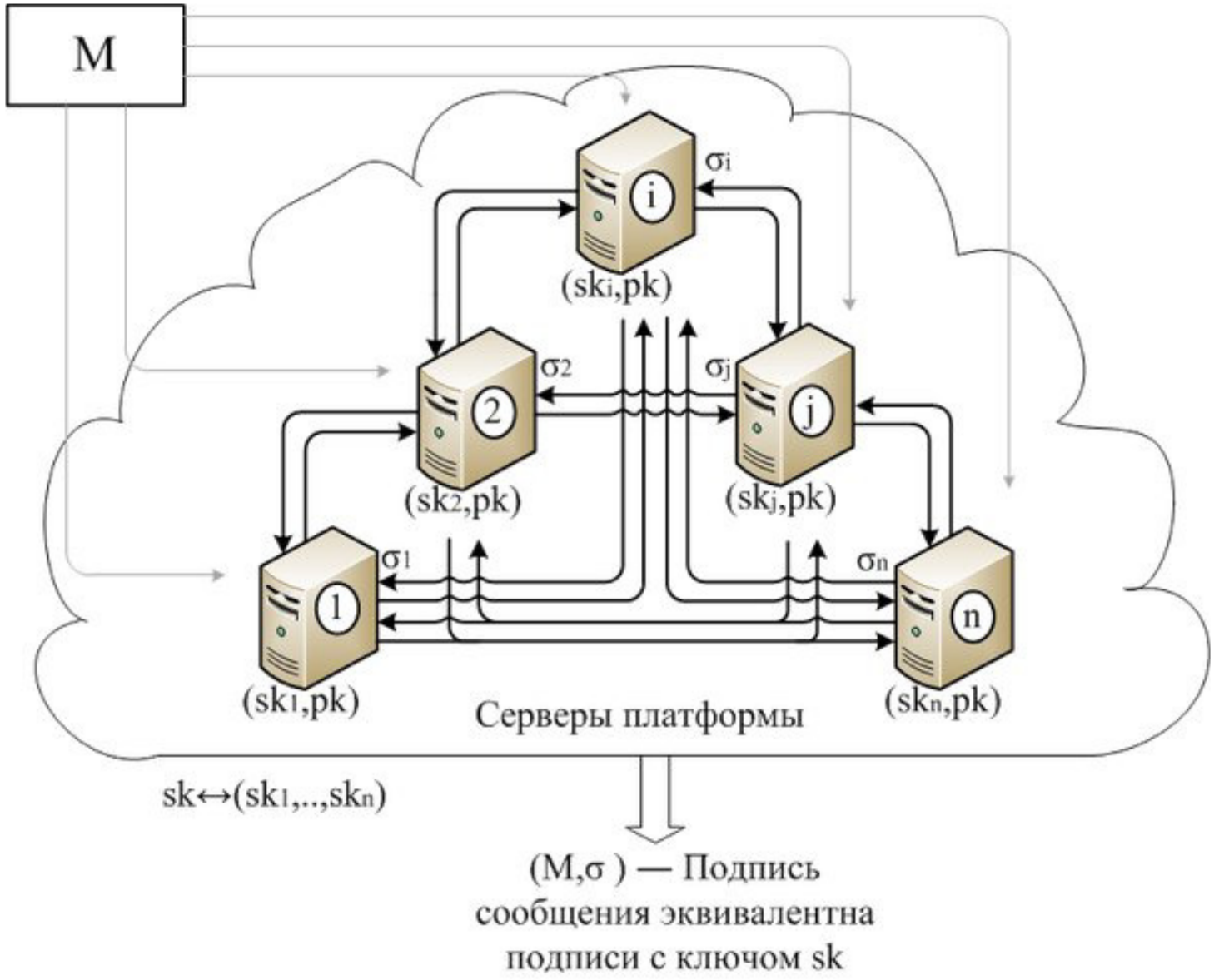
При положительном результате проверки будет выполнено равенство:

$$\left(g^{\delta_0}h^{\gamma_0}\right)\cdot\left(g^{\delta_1}h^{\gamma_1}\right)^{i}\cdot\ldots\cdot\left(g^{\delta_{t-1}}h^{\gamma_{t-1}}\right)^{i^{t-1}} = g^{\delta_0+\delta_1 i+\ldots+\delta_{t-1}i^{t-1}}\cdot h^{\gamma_0+\gamma_1 i+\ldots+\gamma_{t-1}i^{t-1}} =$$

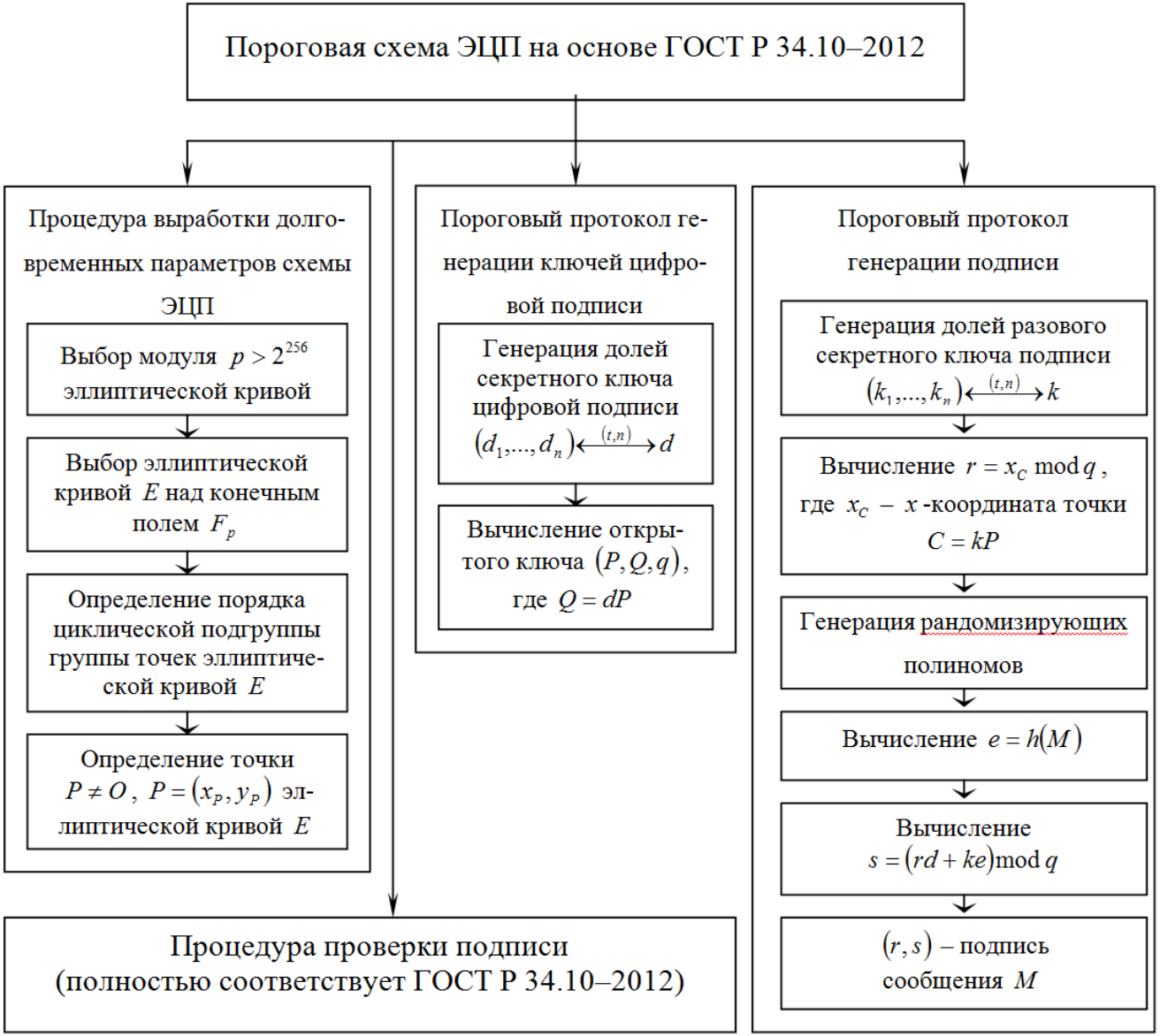
$$= g^{\delta(i)}\cdot h^{\gamma(i)}(\mathrm{mod}\ p).$$

Схема Педерсена обеспечивает теоретико-информационную секретность, так как, даже если вычислительно неограниченный противник, видящий $g^{K}h^{\gamma_0}(\mathrm{mod}\,p)$, умеет решать задачу дискретного логарифмирования и может вычислить $K+d\gamma_0(\mathrm{mod}\,q)$, это все равно не дает ему никакой информации о секрете *K*. Таким образом, схема Педерсена не позволяет противнику вычислить g^{K}, тогда как схема Фельдмана обладает лишь теоретико-сложностной стойкостью относительно знания противником g^{K}.

Пороговая схема цифровой подписи



Пороговая схема подписи на основе ГОСТ Р 34.10-2012 (1)



Пороговая схема подписи на основе ГОСТ Р 34.10-2012 (2)

<table>
<tr><td colspan="3">Пороговый протокол генерации ключей цифровой подписи:
(p,q,m,E,P,t) → ((d1,...,dn),Q)</td></tr>
<tr><td>1</td><td>Генерация долей секретного ключа ЭЦП $d_1,...,d_n$ и открытого ключа ЭЦП Q</td><td>Все узлы S_i, $i=\overline{1,n}$ выполняют протокол совместного разделения случайного секрета на основе СРС Фельдмана. Генерируются доли секрета $(d_1,...,d_n)\xleftrightarrow{(t,n)} d \bmod q$. В результате $\forall S_i$, $i=\overline{1,n}$ получает свою секретную долю d_i общего секрета d, проверочные данные $\{A_l\}$, $l=\overline{0,t}$ и общеизвестную точку эллиптической кривой $Q=dP$. Число d принимается в качестве секретного, а набор (P,Q,q) – в качестве открытого ключа схемы ЭЦП соответственно. G – множество участников, корректно завершивших протокол.</td></tr>
<tr><td colspan="3">Пороговый протокол генерации цифровой подписи:
(p,q,m,E,t,(d1,…,dn),M) → (r,s)</td></tr>
<tr><td>1</td><td>Генерация долей разового секретного ключа ЭЦП $k_1,...,k_n$</td><td>Все узлы $S_i \in G$ выполняют протокол совместного разделения случайного секрета на основе СРС Шамира. Генерируются доли секрета $(k_1,...,k_n)\xleftrightarrow{(t,n)} k \bmod q$. В результате $\forall S_i, i=\overline{1,n}$ получает свою секретную долю k_i общего секрета k.</td></tr>
</table>

Пороговая схема подписи на основе ГОСТ Р 34.10-2012 (3)

<table>
<tr><td rowspan="3">2</td><td rowspan="3">Вычисление $r = x_C \bmod q$, где x_C – x-координата точки $C = kP$</td><td>2.1</td><td>Каждый $S_i \in G$ вычисляет $C_i = k_i P$, рассылает всем остальным узлам сообщение:
$S_i \to S_j,\ j = \overline{1,n},\ j \neq i : [C_i]$
и принимает C_j от других узлов.</td></tr>
<tr><td>2.2</td><td>Каждый $S_i \in G$ выполняет локально интерполяцию (алгоритм Б.5.2): $C = (x_C, y_C) = \text{Interpolate\_ECPoint}(C_{i_1},...,C_{i_{t+1}})$.</td></tr>
<tr><td>2.3</td><td>Каждый $S_i \in G$ вычисляет локально $r = x_C \bmod q$.</td></tr>
<tr><td>3</td><td>Генерация случайных полиномов с нулевым свободным членом</td><td colspan="2">Все узлы $S_i \in G$ выполняют протокол совместного «разделения нуля» на основе СРС Шамира. Генерируются доли секрета $(c_1,...,c_n) \xleftrightarrow{(t,n)} 0 \bmod q$. В результате $\forall S_i$, $i = \overline{1,n}$ получает свою секретную долю c_i, такую, что общий «секрет» $c = 0$.</td></tr>
</table>

Пороговая схема подписи на основе ГОСТ Р 34.10-2012 (4)

4	Вычисление $e = h(M)$	Каждый $S_i \in G$ вычисляет локально *хэш-код сообщения* $e = h(M)$ по *ГОСТ Р 34.11–2012*. Если $e \bmod q \equiv 0$, полагает $e = 0^{255}1$.	
5	Вычисление $s = (rd + ke) \bmod q$	5.1	Каждый $S_i \in G$ вычисляет локально $s_i = (rd_i + k_i e + c_i) \bmod q$, т.е. одну из долей разделенного секрета $(s_1, \ldots, s_n) \xleftarrow{(t,n)} (rd + ke) \bmod q$.
		5.2	Каждый $S_i \in G$ уничтожает долю разового секретного ключа подписи k_i.
		5.3	Каждый $S_i \in G$ рассылает s_i всем остальным узлам: $S_i \to S_j,\ j \neq i : [s_i]$ и принимает s_j от других узлов.
		5.4	Каждый $S_i \in G$ выполняет локально интерполяцию: $s \bmod q = \text{Interpolate}(s_{i_1}, \ldots, s_{i_{t+1}})$.
		5.5	Если $s = 0$, участники возвращаются к шагу (1).
6	Выдача результата	(r, s) – подпись сообщения M.	

Протокол совместного разделения случайного секрета на основе СРС Шамира

Участники: $G = \{S_1,...,S_n\}$ – множество участников, получающих доли секрета.

Вход:
Открытые параметры:
q – простое число, Z_q – поле с определенными в нем операциями модульного сложения и умножения;
n – число участников схемы, t: $t < n$ – порог схемы – максимально допустимое количество скомпрометированных участников.

Выход:
Секретные величины:
x_i, $i = \overline{1,n}$ – полиномиальные доли секрета, переданные соответствующим участникам S_i;
z_i, $i = \overline{1,n}$ – аддитивные доли секрета, переданные соответствующим участникам S_i;
$x \in Z_q$ – значение разделенного секрета, равномерно распределенное в Z_q;
$x \xleftrightarrow{(t,n)} (x_1,...,x_n)$ – условное обозначение (t,n)-порогового разделения секрета.

Описание шагов протокола:

1. Каждый участник S_i в качестве дилера выполняет *протокол разделения случайного секрета* z_i:

1.1) S_i выбирает многочлен f, $\deg f = t$ над полем Z_q: $f_i(z) = \sum_{l=0}^{t} a_{il} z^l \bmod q$, где $a_{i0} = z_i$, а коэффициенты $\{a_{i1}, a_{i2},...,a_{it}\} \in Z_q$ выбираются случайно и независимо;

1.2) S_i вычисляет доли секрета $s_{ij} = f_i(j) \bmod q$, $j = \overline{1,n}$ (в том числе и долю для самого себя) и по каналам связи, обеспечивающим секретность, пересылает их всем другим участникам – держателям долей:

$$S_i \to S_j: \lfloor s_{ij} \rfloor, \ j = \overline{1,n}.$$

2. Каждый участник S_i, получив доли секрета от всех других участников, вычисляет свою полиномиальную долю секрета $x_j = \sum_{S_i \in G} s_{ij} \bmod q$. Значение секрета x не вычисляется в явном виде никем из участников, но неявно оно равно $x = \sum_{S_i \in G} z_i \bmod q$.

Протокол совместного разделения случайного секрета на основе СРС Фельдмана (1)

Участники: $G = \{S_1,...,S_n\}$ – множество участников, получающих доли секрета.

Вход:

Открытые параметры:

p, q – простые числа, такие, что $p = kq + 1$, где $k \in N$; Z_q – поле с определенными в нем операциями модульного сложения и умножения;

E – эллиптическая кривая, $m = lq$, где $l \in N$ – порядок группы точек эллиптической кривой,

$P = (x_P, y_P)$ – точка эллиптической кривой, такая, что $P \neq O$, $qP = O$ – образующий элемент подгруппы порядка q аддитивной группы точек эллиптической кривой;

n – число участников схемы, t: $t < n$ – порог схемы – максимально допустимое количество скомпрометированных участников.

Выход:

Секретные величины:

x_i, $i = \overline{1,n}$ – полиномиальные доли секрета, переданные соответствующим участникам S_i;

z_i, $i = \overline{1,n}$ – аддитивные доли секрета, переданные соответствующим участникам S_i;

$x \in Z_q$ – значение разделенного секрета, равномерно распределенное в Z_q;

$(x_1,...,x_n) \xleftrightarrow{(t,n)} x \bmod q$ – условное обозначение (t,n)-порогового разделения секрета.

Открытые величины:

G – множество участников, корректно завершивших протокол;

$Q = xP$, Q_l, $l = \overline{1,t}$ – проверочная информация для долей разделенного секрета;

$F_{x_i}(z) = f_{x_i}(z) \cdot P$, $S_i \in G$ – проверочные функции для коэффициентов многочленов, выбранных участниками, которые неявно определяют общую проверочную функцию $F_x(z) = f_x(z) \cdot P = \left(\sum_{S_i \in G} f_{x_i}(z) \right) \cdot P$.

Протокол совместного разделения случайного секрета на основе СРС Фельдмана (2)

Описание шагов протокола:

1. Каждый участник S_i в качестве дилера выполняет *протокол разделения случайного секрета* z_i:

1.1) S_i выбирает многочлен f, $\deg f = t$ над полем Z_q: $f_{x_i}(z) = \sum_{l=0}^{t} a_{il} z^l \bmod q$, где коэффициенты $\{a_{i1}, a_{i2}, ..., a_{it}\} \in Z_q$ выбираются случайно и независимо, обозначает $z_i = a_{i0}$, вычисляет точки $Q_i = A_{i0} = a_{i0}P$ и $A_{ik} = a_{ik}P$. S_i рассылает всем остальным участникам открытую проверочную информацию:

$$S_i \to S_j,\ j = \overline{1,n},\ j \neq i:\ \left[A_{ik}, k = \overline{0,t}\right].$$

S_i вычисляет доли секрета $s_{ij} = f_{x_i}(j) \bmod q$, $j = \overline{1,n}$ (в том числе и долю для самого себя) и по каналам связи, обеспечивающим секретность, пересылает их всем другим участникам – держателям долей:

$$S_i \to S_j,\ j = \overline{1,n}:\ \lfloor s_{ij} \rfloor;$$

Протокол совместного разделения случайного секрета на основе СРС Фельдмана (3)

1.2) каждый участник S_j проверяет доли, присланные другими участниками, используя проверочные уравнения:

$$s_{ij} P \stackrel{?}{=} \sum_{k=0}^{t} j^k A_{ik} = F_{x_i}(j),\ i = \overline{1,n},\ i \neq j.$$

Если для какого-либо i уравнение не выполнено, S_j рассылает «обвинение» против S_i.

2. Определяется множество участников G, корректно завершивших протокол:

- если S_i получил более t «обвинений», то он, очевидно, скомпрометирован – и тогда полагаем: $G \leftarrow G \setminus \{S_i\}$;
- если S_i получил от 1 до $t-1$ «обвинений», то он разглашает всем участникам доли s_{ij}, разосланные всем таким S_j, от которых получено «обвинение». Если для каких-либо значений j уравнение не выполнено, то полагаем $G \leftarrow G \setminus \{S_i\}$.

3. Каждый участник S_j вычисляет открытую точку $Q = \sum_{S_i \in G} Q_i$ и свою полиномиальную долю секрета: $x_j = \sum_{S_i \in G} s_{ij} \bmod q$. Значение секрета x не вычисляется в явном виде никем из участников, но неявно оно равно $x = \sum_{S_i \in G} z_i \bmod q$.

Протокол совместной генерации случайного многочлена с нулевым свободным членом на основе СРС Шамира

Участники: $G = \{S_1,...,S_n\}$ – множество участников, получающих доли секрета.

Вход:

Открытые параметры:

q – простое число, Z_q – поле с определенными в нем операциями модульного сложения и умножения;

n – число участников схемы, t: $t < n$ – порог схемы (максимально допустимое количество скомпрометированных участников).

Секретные величины:

x_i, $i = \overline{1,n}$ – полиномиальные доли секрета каждого участника S_i.

Выход:

Секретные величины:

$\tilde{x}_i$, $i = \overline{1,n}$ – полиномиальные доли секрета, переданные соответствующим участникам S_i;

$z_i = 0$, $i = \overline{1,n}$ – аддитивные доли секрета, переданные соответствующим участникам S_i, такие, что $\sum_{S_i \in G} z_i = 0 \bmod q$

Описание шагов протокола:

1. Каждый участник S_i в качестве дилера выполняет протокол разделения секрета $z_i = 0$:

1.1) S_i выбирает многочлен f, $\deg f = t$ над полем Z_q: $f_i(z) = \sum_{l=0}^{t} a_{il} z^l \bmod q$, где $a_{i0} = z_i$, а коэффициенты $\{a_{i1}, a_{i2},...,a_{it}\} \in Z_q$ выбираются случайно и независимо;

1.2) S_i вычисляет доли секрета $s_{ij} = f_i(j) \bmod q$, $j = \overline{1,n}$ (в том числе и долю для самого себя) и по каналам связи, обеспечивающим секретность, пересылает их всем другим участникам – держателям долей:

$$S_i \to S_j,\ j = \overline{1,n}: \lfloor s_{ij} \rfloor.$$

2. Каждый участник S_i, получив доли секрета от всех других участников, вычисляет свою полиномиальную «долю нуля»: $x_j^* = \sum_{S_i \in G} s_{ij} \bmod q$. Значение секрета $x = \sum_{S_i \in G} a_{i0} = 0$.

Новые полиномиальные доли секрета каждого участника равны $\tilde{x}_j = x_j + x_j^* (\bmod q)$.

Процедура интерполяции по Лагранжу

Участники: выполняется локально.

Вход:

q – простое число, Z_q – поле с определенными в нем операциями модульного сложения и умножения;

$\{x_1,...,x_n\}$ – множество долей секрета, полученных от участников схемы разделения секрета $\{S_1,...,S_n\}$, из которых не более t могут отсутствовать, причем $n \geq 2t+1$, а оставшиеся удовлетворяют условию $x_i = f(i) \bmod q$, где f – некоторый многочлен, такой, что $\deg f = t$;

$Q \subseteq \{1,2,...,n\}$ – множество номеров участников схемы, доли которых удовлетворяют указанному условию и взяты для восстановления секрета.

Выход:

x – секретное число, которое было распределено между участниками $\{S_1,...,S_n\}$;

$x=\mathrm{Interpolate}(x_1,\ldots,x_n)$ – условное обозначение величины, восстановленной при помощи настоящей процедуры.

Описание процедуры:

Вычислить по интерполяционной формуле Лагранжа

$$f(x) = \sum_{j=1}^{t} \lambda_j f(j), \text{ где } \lambda_j = \prod_{\substack{k \in Q, \\ k \neq j}} \frac{x-k}{j-k}$$

(считая, что участникам раздаются значения многочлена в точках, соответствующих их условным порядковым номерам). Так как разделенный секрет представляет собой значение многочлена в нулевой точке, формула допускает упрощение:

$$x = f(0) = \sum_{j \in Q} \lambda_j x_j, \text{ где } \lambda_j = \prod_{\substack{k \in Q, \\ k \neq j}} \frac{k}{k-j},$$

а коэффициенты интерполяции по формуле Лагранжа вычисляются предварительно.

Проактивная криптография

Далее пороговая схема ЭЦП преобразуется к модели «проактивной безопасности». Для этого все время работы системы разбивается на *циклы*, не обязательно одинаковой продолжительности. В начале каждого цикла выполняется фаза обновления секретных ключей. Весь остальной период времени система функционирует в обычном режиме. Секретный ключ ЭЦП разделяется между узлами с использованием схемы проверяемого разделения секрета. В каждой фазе обновления доли ключа заменяются новыми, но ключ в целом не изменяется. Таким образом, вся секретная информация с узлов, собранная противником за промежуток времени между двумя последовательными фазами обновления секретных ключей, оказывается бесполезной для него в следующем цикле работы системы. Теперь условие, состоящее в том, что противник может скомпрометировать не более *t* узлов, применяется к каждому циклу работы системы в отдельности. Таким образом, в течение всего ЖЦ каждый узел может быть скомпрометирован, но за время одного цикла может быть скомпрометировано не более *t* любых узлов.

Обнаружение несанкционированного применения секретных ключей (1)

Механизм 1. **Наблюдение за любым отдельным сообщением в потоке сообщений, вне зависимости от предыдущих событий.**

Механизм 2. **Наблюдение за сообщениями в том случае, когда данная последовательность сообщений не может произойти при санкционированном использовании секретов. Этот механизм требует достаточной информации о предыдущих сообщениях, чтобы определить, противоречат ли им новые наблюдения.**

Механизм 3. **Наблюдение за сообщениями в том случае, когда сообщения противоречат знаниям участников диалога о своей собственной деятельности. Для этого требуется достаточно информации о предыдущих действиях участников, а также о предыдущих сообщениях, чтобы определить, кто отправлял текущие наблюдаемые сообщения.**

Обнаружение несанкционированного применения секретных ключей (2)

Комбинация трех механизмов обнаружения активной компрометации секретов приводит к ряду принципов проектирования для обнаружения злоупотребления ключами.

Принцип 1. **Сообщения протокола должны быть связаны с предыдущими сообщениями (к примеру, можно использовать счетчик и при каждом отправляемом сообщении увеличивать его). Это помогает максимизировать возможность обнаружения компрометации секрета и предотвращает «рассинхронизацию» после несанкционированного использования секретов.**

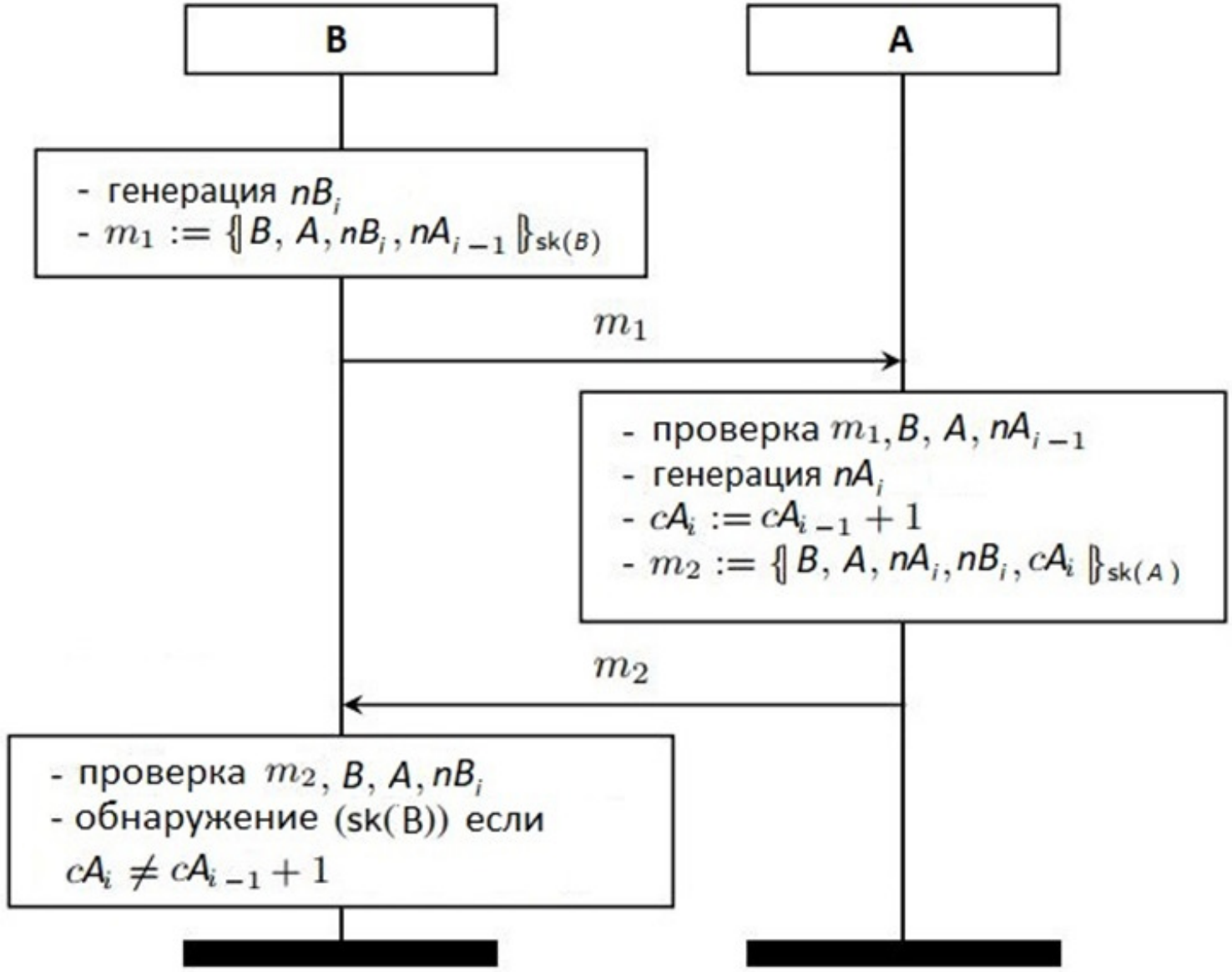
Принцип 2. **Включение уникальных и непредсказуемых значений в сообщения. Это помогает установить противоречивые наблюдения и гарантировать, что противник не сможет правильно предсказать, что сделает клиент в дальнейшем. Этот принцип важен, так как если противник может предсказать следующий обмен сообщениями, он потенциально может провести его заранее с одним из участников, а затем занять свое место в реальном обмене, не оставив никаких доказательств.**

Обнаружение несанкционированного применения секретных ключей (3)

Принцип 3. **Максимальное распространение данных, которые другие стороны могут найти противоречивыми. Обнаружение требует наблюдений, поэтому важно увеличить возможности для этого. В идеальном случае некоторые наблюдения могут быть переданы всем участникам, но не для всех приложений это возможно.**

Принцип 4. **Определение клиентов, которые являются инициаторами обмена, и уверенность в том, что они могут наблюдать за сообщениями. Инициаторы могут обнаружить больше, чем клиенты, которые могут обнаруживать несанкционированное использование секретов только путем наблюдения противоречий. Например, в настройке PKI инициирующими клиентами являются владельцы домена и удостоверяющего центра, поскольку сертификат для домена, подписанного ключом удостоверяющего центра, должен быть создан только после того, как он был запрошен доменом и подписан этим центром сертификации. Если такой сертификат встречается без запроса или без его подписания, то ключ использован несанкционированно.**

Протокол-счетчик (1)



Протокол-счетчик (2)

В протоколе один участник обмена увеличивает счетчик каждый раз, когда другой предоставляет новую подпись. Принцип счетчика может использоваться для обнаружения любого закрытого ключа, но в данном разделе он иллюстрируется на примере протокола, в котором происходит генерация подписи. Значение счетчика увеличивается не более одного раза для каждого использования подписанного ключа.

Сообщение *m*, подписанное с помощью закрытого ключа *sk*, представлено как значение $\{|m|\}_{sk}$, включает как подпись на *m*, так и открытый текст сообщения *m*, cA_i — *i*-е значение счетчика клиента *A*, cB_i — *i*-е значение счетчика клиента *B*, nA_i — *i*-е значение *nonce* клиента *A*, nB_i — *i*-е значение *nonce* клиента *B*.

Протокол-счетчик требует, чтобы клиент *A* увеличивал счетчик один раз для каждого уникального подписанного сообщения клиента *B*, также требуя от *B* использовать *nonce*, сгенерированное *A* во время предыдущего сеанса. Таким образом, *B* может определить, было ли вызвано приращение счетчика пользователем *A*, сравнив счетное состояние с последним состоянием, возвращенным *A*.