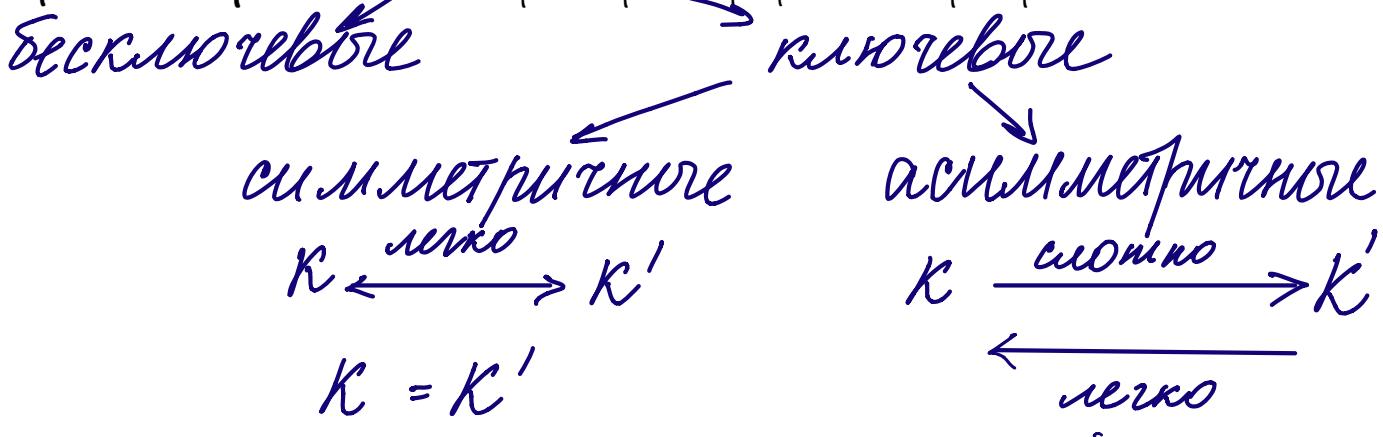


# Криптографические методы защиты информации.

## I. Основные понятия криптографии с открытым ключом. Принципы построения крипtosистем с открытым ключом

Криптографический приметив - атомарное криптографическое преобразование.



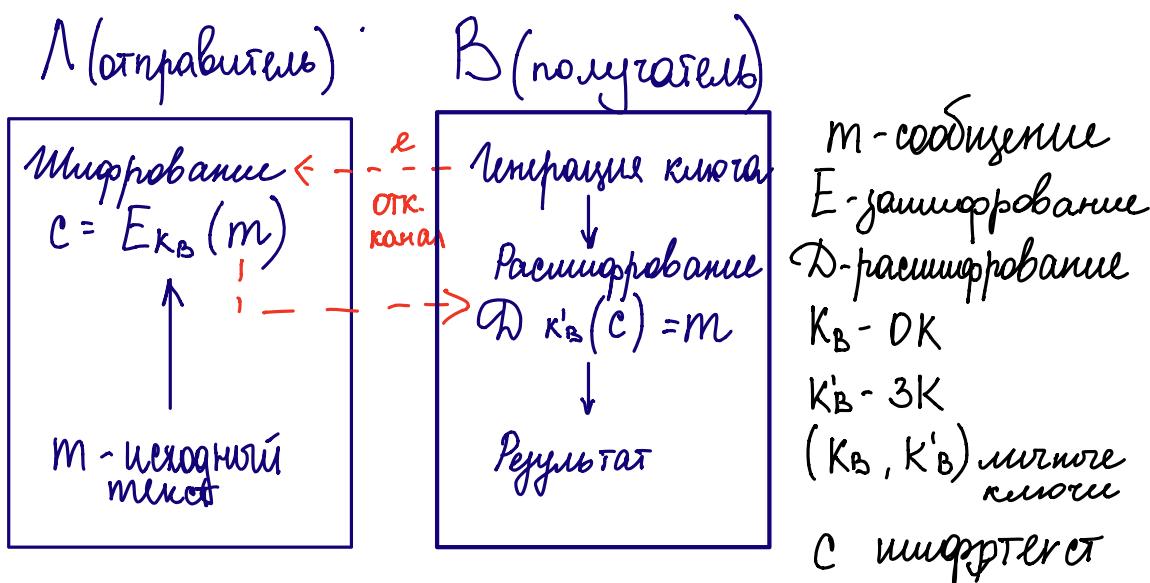
Криптографическая система с открытым ключом - система шифрования или/и электронная подпись, при которой открытый ключ передаётся по открытому каналу и используется для проверки ЭП и шифрования сообщений.

Алгоритмы шифрования с открытым ключом решают задачи, возникшие при использовании симметричной криптографии:

- распределение ключа
- Необходимость цифровой подписи

Для генерации ЭП и расшифрования сообщения используется **закрытый ключ**.

### Схема криптографического преобразования



## Математические задачи

$$\textcircled{1} \quad K = f(K)$$

$$OK = f(CK)$$

$f$  должна быть одноточечной в  
одну сторону. (односторонняя)

$$f: X \rightarrow Y$$

Свойства:

- 1)  $\exists$  полиномиальный алгоритм вычисления  $f(x)$
- 2)  $\nexists$  полиномиального алгоритма вычисления  $f^{-1}(y)$

$$\textcircled{2} \quad f(x) = a^x \pmod p$$

$f$  функция с лацканом

$$f_k: X \rightarrow Y$$

Свойства:

- 1)  $\exists$  полином. алгоритм вычисления  $f_k(x) \quad \forall k, x$
- 2)  $\nexists$  полином. алгоритма вычисления  $f_k(x)$  при неизвестном  $k$ .
- 3)  $\exists$  полином. алгоритм вычисления  $f_k(y)$  при известном  $k$ .

Пример: задача разложения на множители

$$x^2 \equiv a \pmod{pq} \text{ разрешено} \Leftrightarrow \left(\frac{a}{p}\right) = \left(\frac{a}{q}\right) = 1$$

$$n = pq$$

$$x^2 \equiv a \pmod{n}$$

$$c \equiv m^2 \pmod{n}$$

$$y = f(x) = x^2 \pmod{n}$$

Если знать  $p$  и  $q$ , то можно восстановить  $x$ .

# Основные криптографические примитивы

## 1. Шифр

$A \rightarrow B$       A шифрует на РК  
B расшифров. на SK

## 2. Цифровая подпись

- ✓ обеспечивает неворовство отката от авторства
- ✓ неоспоримая подпись
- ✓ групповая подпись

- подпись с восстановлением сообщения

$$m - \text{сообщение} \quad s \leftarrow \text{sign } \text{SK}_A(m) \quad m' \leftarrow \text{Verify } \text{PK}_A(s) \quad m = m'?$$

- подпись без восстановления сообщения

$$\boxed{m \mid s} \quad s \leftarrow \text{Sign } \text{SK}_A(m) \quad \text{Verify } \text{PK}_A(m, s) = \begin{cases} \text{True} \\ \text{False} \end{cases}$$

## 3. Хэш-функция

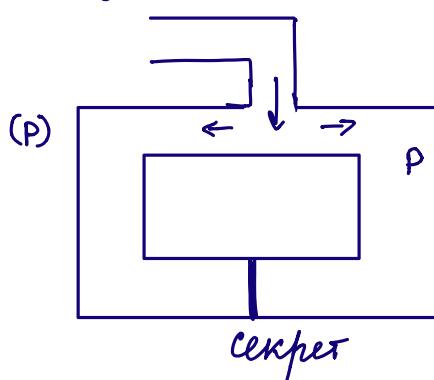
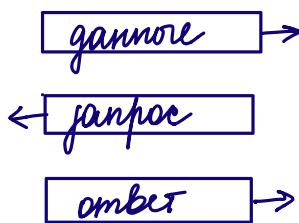
$m \rightarrow h(m)$  ✓ преобразование данных произвольной длины в фиксир. длину  
✓ свободна от коллизии

Коллизия I рода:  $m, m' \quad h(m) = h(m')$   $(m, s) \rightarrow (m', s')$

Коллизия II рода:  $m, h(m) \rightarrow m'$   $h(m') = h(m)$   
(внешний парушийтель)

## 4. Доказательство с нулевым разглашением

$b = a^x \xleftarrow{\text{c.к.}}$  Необходимо доказать, что знаем  $x$  и не разглашаем его доказ.



## Стойкость криптографических протоколов

1. Безусловно стойкие  
шифр Вернама
2. Доказано стойкие  
задача дискр. логарифм-а, RSA, DH
3. Предположительно стойкие  
основаны на частной матем. задаче

## Основные возможности нарушителя

1. Вычислительные возможности (оперативная память)
2. Математические возможности (разработка методов и алг-ов)
3. Криптографические возможности (частная матем. задача)
4. Лабораторные возможности (временные атаки)

## 2. Крипtosистема RSA: выбор параметров, шифрование с открытым ключом. Задача RSA. Связь задачи RSA с задачей разложения на множители.

### Выбор параметров

Корректность параметров связана с оценкой стойкости системы и может быть определена лишь с точки зрения практической стойкости. Следовательно, корректные параметры должны быть построены так, чтобы минимизировать ущерб от известных подходов к ослаблению крипtosистемы. Прежде всего, следует учитывать, что слабость одного из параметров практически не компенсируется усилением свойств других параметров.

Число  $n = pq$  должно быть большим. Числа  $p, q$  не должны содержаться в списках известных больших простых чисел, не должны быть слишком близки друг к другу, либо существенно различаться по величине. Они не должны быть построенными по детерминированным алгоритмам с небольшим числом

известных вариантов начальных параметров или содержать закономерности в двоичной записи. В общем,  $p$  и  $q$  не должны отличаться от типичных представителей случайных простых чисел. Аналогичными свойствами должны обладать параметры  $e$  и  $d$ . Например, если секретный ключ  $d$  содержит в двоичной записи небольшое количество единиц, то номера мест этих единиц, легко определить перебором. Можно доказать, что при известном  $d$  существует возможность факторизации модуля. Известно, что для чтения сообщений, зашифрованных криптосистемой RSA, достаточно знания некоторого кратного функции Эйлера от модуля, т.к. в этом случае можно вычислить ключ, криптоэквивалентный ключу  $d$ .

Если  $p$ -простое,  $\forall a \neq p \quad a^{p-1} \equiv 1 \pmod{p}$  т. Ферма  
 $q$ -простое,  $\forall a \neq q \quad a^{q-1} \equiv 1 \pmod{q}$

Надо найти  $t$ :  $a^t \equiv 1 \pmod{pq}$   $t$ -наименьшее  
 $\Downarrow t = \text{НОК}(p-1, q-1) \quad t = (p-1)(q-1)$

$n = pq \quad p-1 = 2 \cdot p_1$  или  $q-1 = 2q_1 \quad (p \neq q)$   
 Если  $n = pq \Rightarrow \varphi(n) = (p-1)(q-1) \Rightarrow \varphi(pq) = \frac{\varphi(p)\varphi(q)}{(p-1)(q-1)}$   
 Если  $a^{r-1} \equiv 1 \pmod{p} \quad a^p \equiv a \pmod{p} \quad \forall a$

Теорема Эйлера в теории чисел гласит:

Если  $a$  и  $m$  взаимно просты, то  $a^{\varphi(m)} \equiv 1 \pmod{m}$ , где  $\varphi(m)$  — функция Эйлера.

Важным следствием теоремы Эйлера для случая простого модуля является малая теорема Ферма:

Если  $a$  не делится на простое число  $p$ , то  $a^{p-1} \equiv 1 \pmod{p}$ .

$p, q$ -простые;  $p \neq q$ .  $n = pq$

$\|p\| \sim \|q\|$

$s \in \{[\sqrt{n}], [\sqrt{n}+1], \dots\}$

$n = s^2 - t^2$  чем ближе число  $p$  и  $q$ , тем быстрее проходит разложение на множители

Теорема 1.

Задача разложения на множители и задача вычисления функции Эйлера для  $n=pq$ , где  $p$  и  $q$ -простые, polynomialno решаются.

Если умеем решать 1ю задачу за полиномальное время, то примерно за такое же время решается 2 задача.

DDK-БО:

$$\Rightarrow n = pq \quad \varphi(n) = (p-1)(q-1)$$

$$\Leftarrow \varphi(n) = (p-1)(q-1) = pq - (p+q) + 1 = n - (p+q) + 1$$

$$\begin{cases} n = pq \\ p+q = n - \varphi(n) + 1 \end{cases} \quad \text{но т. Вместе находим корни}$$

$$\Downarrow x^2 - (n - \varphi(n) + 1)x + pq = 0 \quad p, q - \text{корни ур-я} \quad \blacksquare$$

### Шифрование с открытым ключом

OK:  $(e, n)$   $e \in (\mathbb{Z}/\varphi(n)\mathbb{Z})^*$  e - открытий показатель

d - закрытий показатель

СК:  $(d, n)$   $d \equiv e^{-1} \pmod{\varphi(n)}$   $e \cdot d \equiv 1 \pmod{\varphi(n)}$

### Зашифрование

$$m \in \mathbb{Z} / n \mathbb{Z}$$

с - шифртекст

$$c \leftarrow m^e \pmod{n}$$

### Расшифрование

$$c^d = (m^e)^d \stackrel{\text{прижать}}{=} m^{ed} \stackrel{\text{mod}(m, n)=1}{\equiv} m \pmod{n}$$

Числ. подпись

$$s = m^d \pmod{n}$$

d - закр. ключ форм. я подпись

### Формирование систем RSA:

1. Выбираем 2 различные числа  $p$  и  $q$ .
2. Вычислить  $n = pq$  и  $\varphi(n) = (p-1)(q-1)$ .
3. Выбираем число  $e$  взаимопростое с  $\varphi(n)$   $1 < e < \varphi(n)-1$
4. Вычислить число  $d$  (многозначимо обратное к  $e$ ) из уравнения  $d \cdot e \equiv 1 \pmod{\varphi(n)}$
5. Определить открытые ключи  $e$  и  $n$ .
6. Определить закрытые ключи  $d, p, q$  и  $\varphi(n)$ .

Данная схема не исп-тся на практике, т.к. не является практически надежной.  
 $E(m)$  - это 1-те дешифрований

откр.  
пок-но

### Задача RSA

Дано:  $(e, n)$ ,  $c$  ↑  
шифртекст  $\Rightarrow$  в общем случае  $\sqrt{c} \pmod{n}$

Найти:  $m$

### Теорема 2.

Задача RSA не сложнее задачи разложения на множители.

Док-во:

$$\begin{aligned} n &\rightarrow p, q \rightarrow \varphi(n) = (p-1)(q-1) \rightarrow d \equiv e^{-1} \pmod{\varphi(n)} \rightarrow \\ &\rightarrow m = c^d \pmod{n} \quad \blacksquare \end{aligned}$$

3. Разложение составного числа на множители по известным показателям RSA (внутренний и внешний нарушитель). Атака Винера на криптосистему RSA (случай малого секретного показателя)

### Свойство гомоморфности

Крипт-ма RSA обладает св-вом гомоморфности  
 $f(ab) = f(a)f(b)$

$$f(m) = m^e \pmod{n}$$

$$f(m_1)f(m_2) = m_1^e \cdot m_2^e = (m_1m_2)^e = f(m_1m_2) \pmod{n}$$

Свойство гомом-ти лежит в атаке на основе подобранныго шифртекста.

Нарушитель имеет доступ к расшифрованному аппарату и может подавать все, кроме истинного шифртекста.

$C$ - шифртекст

$$x - \text{сугайное } \in (\mathbb{Z}/n\mathbb{Z})^*$$

$$\hat{C} = C \cdot x^e \pmod{n}$$

$$(\hat{C})^d = (C \cdot x^e)^d = (m^e)^d \cdot (x^e)^d = m \cdot x \pmod{n}$$

$$(\hat{c})^d \cdot x^{-1} \equiv m \pmod{n}$$

### Случай малого закрытого показателя

$d$ -достаточно мал, тогда по числу  $n$  и открытым показателю  $e$  можно найти  $d$ .

Можно найти длину закрытого показателя и число единиц в двоичном представлении.

$r = \log_2(d / \sqrt[4]{n}) \Rightarrow$  задача поиска  $d$  и разложение числа  $n$  требует  $2r+8$  бит.

Если  $d < \sqrt[4]{n}$ , то показатель  $d$  можно найти без перебора.

Открытый и закрытый показатели связаны соотношением:  
 $e \cdot d \equiv 1 \pmod{(p-1)(q-1)}$ .  $ed \equiv 1 \pmod{\varphi(n)}$

$$H = \text{НОД}(p-1, q-1); G = \text{НОД}(p-1, q-1) \Rightarrow \exists k \in \mathbb{Z}, \text{ что}$$

$$e \cdot d = 1 + kH = 1 + \frac{k(p-1)(q-1)}{G}, \text{ откуда}$$

$$\frac{e}{n} = \frac{k(p-1)(q-1)}{dGn} + \frac{1}{dn} = \frac{k}{dG} \cdot \frac{n-p-q+1+\frac{G}{k}}{n} = \frac{k(1-\delta)}{dG} = \frac{k(1-\delta)}{dg}$$

$\frac{k}{dg}$  - дробь  $\frac{k}{dG}$  после сокращения, при этом  $d=1$  или  $d=2$

Возьмем  $\frac{k}{dg}$  через  $\delta$  и дробь  $\frac{e}{n}$ , при этом

$$\frac{\sqrt{2}}{\sqrt{n}} < \delta < \frac{4}{\sqrt{n}}. \text{ Далее возвращаются подкодящие}$$

дроби для квадр. дроби  $\frac{4}{\sqrt{n}}$ . Подкодящая дробь со знаменателем ближним к  $\sqrt{n}$  является хорошим приближением к  $\delta$ .

Можно найти показатель  $d$ , если длина достаточно мала.

## Атака Винера

Атака Винера, если  $d \leq \frac{1}{3} \sqrt[4]{n}$ .

$$e \cdot d \equiv 1 \pmod{\varphi(n)}$$

$$|ed - 1| = k\varphi(n)$$

$$\left| \frac{e}{\varphi(n)} - \frac{k}{d} \right| = \frac{1}{d\varphi(n)} \quad \frac{e}{\varphi(n)} \approx \frac{e}{n} \Rightarrow$$

$$\left| \frac{e}{n} - \frac{k}{d} \right| = \left| \frac{ed - kn}{nd} \right| = \left| \frac{ed - kn + 1 - 1}{nd} \right| = \left| \frac{1 - k(n - \varphi(n))}{nd} \right| \leq \frac{3k}{d\sqrt{n}} < \frac{1}{2d^2}$$

$\uparrow$  верно       $\uparrow$  неизвестно

Если существует  $d$  Эвклидова дробь  $\frac{p}{q}$ :  $|d - \frac{p}{q}| < \frac{1}{2q^2}$ , то  $\frac{p}{q}$  подх. дробь к  $e$ .

Получим, что  $\frac{k}{d}$ -подх. дробь  $n$ .

Число подходящих дробей конечно. Последняя  $\frac{k}{d}$ .

Алгоритм атаки:

Вход: число  $n$ , показатель  $e$ .

Выход: закр. подх. дробь  $d$ .

1. Представить  $\frac{e}{n}$  в виде непрерывной дроби  $[0; d_1, d_2, \dots, d_m]$ .
2. Для  $i = 1, \dots, m$ :
  - 2.1. Вычислить подходящую дробь  $\frac{p_i}{q_i}$ .
  - 2.2. Проверить выполнение сравнения  $(m^e)^{q_i} \equiv m \pmod{n}$ ,  
где  $m$ -произв. сообщение.
3. Если сравнение выполнено, то  $d = q_i$ .

## Случай специальных отк.-показателей

Кандидаты показ-ль и система делителей имеют персональное число  $n_i = p_i q_i$ .

Для ускорения используют малые отк.-показатели  $e_i$ .

Здесь если  $n_i$  различны, можно выполнить бесконечное цешифрование сообщение.

Бескодное дешифрование шифрования-кода сообщения в случае малого общего пок-ия  $e$ .

Вход: открытое тексты  $(e, n_1), (e, n_2), \dots, (e, n_z)$ , шифртексты шифр-ных сообщений:  $c_1, c_2, \dots, c_z$ .

Выход: сообщение  $m$ .

1.  $\exists c_1, c_2, \dots, c_z$  и характеристики  $n_1, n_2, \dots, n_z$  такие что  $kT \in \mathbb{Z}$  восстановить  $x = m^e$ .
2. Восчислить  $m = \sqrt[e]{x} \in \mathbb{Z}$
3. Рез-м:  $m$ .

Атака на основе "частично известных" От

$M_1$  и  $M_2$  не одинаковы, следовательно есть некоторое аффинное соотношение с куб. параметрами  $M_2 \equiv dM_1 + \beta \pmod{n}$ , где  $d, \beta$  известны, то по шифртекстам  $c_1, c_2$  можно восстановить два сообщения.

Сообщение  $M_2$  восчисляется через  $M_1$  и квад-мн  $d, \beta$ . В случае правильного  $e$ :

$$m_1^e - c_1 \equiv 0 \pmod{n}, (dm_1 + \beta)^e - c_2 \equiv 0 \pmod{n}$$

$m_1, m_2$   
 $M_2 = dm_1 + \beta$   
 $d, \beta$  известны

Известное  $M_1$  будет  $Z$ . Тогда, переходя к концу полиномов  $(\mathbb{Z}/n\mathbb{Z})$  можно записать:

$$\text{НОД}(Z^e - c_1, (dm_1 + \beta)^e - c_2) = Z - M_1 \in (\mathbb{Z}/n\mathbb{Z})$$

Восчислив НОД, можно найти  $m_1$ .

Алгоритм

Вход:  $n, c_1, c_2, d, \beta, e$

Выход:  $m_1, m_2$

1. Восчислить  $m_1$   $Z - M_1 = \text{НОД}(Z^e - c_1, (dm_1 + \beta)^e - c_2)$
2.  $M_2 \leftarrow dm_1 + \beta \pmod{n}$
3. Рез-м:  $(m_1, m_2)$

## 4. Атаки на криптосистему RSA.

### 1. Сургай обидно модуль

В системе RSA у двух полуводателей общий составное число  $n$ , но разные открытые ключи  $e_1, e_2$  и закрытые ключи  $d_1, d_2$  такие, что  $e_1 d_1 = e_2 d_2 = 1 \pmod{\varphi(n)}$ , то какому полуводателю может найти разложение числа  $n$  и узнать закрытый ключ другого полуводателя.

Для этого надо найти значение квадратичной единицы, отличное от  $\pm 1$ .  
если  $t^2 \equiv 1 \pmod{n}$ ,  $t \neq \pm 1$ , то  $t^2 - 1 \equiv (t+1)(t-1) \equiv 0 \pmod{n}$  и  
 $t+1$  и  $t-1$  имеют неприводимые делители с  $n$ .

Алгоритм разложения составного числа на мн-ца по известным показателям RSA.

Вход:  $n, e, d$ , то  $ed \equiv 1 \pmod{\varphi(n)}$

Выход: делители  $p$  и  $q$  числа  $n$ .

1.  $N \leftarrow ed - 1$ .

представить  $n$  в виде  $N = 2^s S$ ,  $s$ -нечетное число

2. Выбрать случайное  $a$  и выполнить  $b \leftarrow a^s \pmod{n}$ .

3. Вычислить  $b^{2^0} \equiv b \pmod{n}$

$$b^{2^1} \equiv (b^{2^0})^2 \pmod{n}$$

$$b^{2^2} \equiv (b^{2^1})^2 \pmod{n}$$

до тех пор, пока  $b^{2^m} \equiv 1 \pmod{n}$

если  $b^{2^{m-1}} \equiv -1 \pmod{n}$ , то к ②, иначе  $t \leftarrow b^{2^{m-1}} \pmod{n}$

4. Выполнить  $p \leftarrow \text{НОД}(t+1, n)$ ,  $q \leftarrow \text{НОД}(t-1, n)$

5. Рез-м:  $(p, q)$

## 5. Схема цифровой подписи RSA. RSA-PSS. Атаки на схему подписи RSA. Ошибка в реализации. Временная атака.

### Подпись RSA

Формирование подписи:  $s \leftarrow m^{d_a} \pmod{n_a}$

Проверка подписи:  $m \leftarrow s^{e_a} \pmod{n_a}$

- Недостатки:
- 1) сообщение ограничено  $m \in \mathbb{Z}/n\mathbb{Z}$
  - 2) св-во гомоморфности для нарушимеля

Получив пару  $(m', s')$  без ключа нарушиль проходит проверку подписи.

### Действие нарушиля:

1. Выбирается случайное  $d \in (\mathbb{Z}/n\mathbb{Z})^*$
  2. Вычисляется  $y \leftarrow d^{e_a} \pmod{n_a}$ .
- Формированием  $(m', s')$  путем передачи на подпись, а не путем сформированием данных.
3.  $m \leftarrow y \cdot m' \pmod{n_a}$   $m$  - на подпись
  - Участник A (подписывающий)
  4.  $s \leftarrow m^{d_a} \pmod{n_a}$
  - $(m, s)$  - пара, подписанная на клюе участников A.
  5.  $s' \leftarrow s \cdot d^{-1} = (m')^{d_a} \pmod{n_a}$

$(m', s')$  участник подписал, в р-те  
злоум-к получил пару  $(m', s')$ .

Напрочь RSA использовать неизв.

### RSA-PSS (probabilistic signature scheme)

вероятностной

Асимметричный алгоритм подписи, основанный на принципе кодирования RSS. Был разработан, чтобы обеспечить современные методы анализа безопасности.

Выход доп. обработка:  $D = \lceil \log_2 n \rceil$  листаце членов сверху

$$k_0, k_1 \in \mathbb{Z} : k_0 + k_1 \leq D-1$$

$k_1$  - длина алг-та  
 $k_0$  - битов, которые остались

$$G: \{0,1\}^{k_1} \rightarrow \{0,1\}^{D-k_1-1}$$

$$H: \{0,1\}^* \rightarrow \{0,1\}^{k_1} \quad k_1 = 256, 512 \text{ бит}$$

$$G_1: \{0,1\}^{k_1} \rightarrow \{0,1\}^{k_0}$$

$$G_2: \{0,1\}^{k_1} \rightarrow \{0,1\}^{D-1-k_1-k_0}$$

вероятность близка к  
нахождению обратной  
ф-ции RSA.

## Формирование подписи:

1. формируется строка  $r \in \{0,1\}^{k_0}$
2. Вычисление  $W \leftarrow M(m \parallel r)$
3.  $\parallel$  до пары модуля  
 $M \leftarrow \emptyset \parallel W \parallel G_1(W) \oplus r \parallel G_2(W)$
4.  $S \leftarrow M^d \pmod{n}$

## Проверка подписи:

1.  $M \leftarrow S^e \pmod{n}$
2.  $M = b \parallel W \parallel d \parallel B$
3.  $G(W) = G_1(W) \parallel G_2(W)$  } вычисление
4.  $r = G_1(W) \oplus d$
5.  $b = ?$   
 $G_2(W) = ?$   
 $M(m \parallel r) = ?$  } проверки

## Атаки на схему подписи

1. Для разных составляющих модуля разные процедуры.

$$d_p \equiv d \pmod{p-1} \quad d_q \equiv d \pmod{q-1}$$

\* Если  $p$  и  $q$  имеют неодинаковой вид

$$\begin{cases} S_p \equiv M^{d_p} \pmod{p} & \text{нечетной } p\text{-м} \\ S_q \equiv M^{d_q} \pmod{q} & \text{четной } p\text{-м} \end{cases}$$

$p$  и  $q$  могут быть  
различными

$$\begin{aligned} S &\rightarrow S + z \cdot q \\ S &\not\equiv S + z \cdot q \pmod{p} \end{aligned}$$

$$(m, \underbrace{s + z \cdot q}_{S'}) \quad (S')^e \cdot \text{вычислить}$$

$$\text{НОД}((S')^e - m, n) = \text{НОД}((S + z \cdot q)^e - m, n) = \text{НОД}(C_e s^e (zq)^{e-i}, n) = q$$

## 2. Временная атака

$$d = (1 \dots 1)_2$$

$$M_i^2 \pmod{n}, M_i^2 M_i \pmod{n}$$

Часто интересно смотрят сколько времени тратится на базовые операции.

$$M_i^d \pmod{n} \quad \left\{ \begin{array}{l} T_1 - \text{время на формирование} \\ \text{подписи.} \end{array} \right.$$

Следующее выполнение коррелирует

$$\begin{aligned} M_i^2 M_i \pmod{n} &\rightarrow 1 \\ M_i^2 \pmod{n} &\rightarrow 0 \end{aligned}$$

При реализации:

1. явисимость 0 и 1 должна быть минимальной
2. искусственно замедлить возведение в степень

## 6. Концепция подписи вслепую. Схема подписи вслепую на основе RSA

Подпись вслепую - подпись, особенностью которой является то, что подписывающая сторона не может знать содержимое сообщения.

Числ: воспринимать сообщение, которое он знакомиться с сообщением стороны A, которое он подписывает, и с соответствующей подписью под этим сообщением.

Потому в дальнейшем подписанное сообщение невозможно связать со стороной A.

Пример:

- 1) подпись билотеней на участках
- 2) электронные патенты

### Банковские системы

Наиболее широкое применение протокол слепых подписей нашло в сфере [цифровых денег](#).

Например, чтобы вкладчик не обманул банк, может использоваться такой протокол: вкладчик пишет одинаковый номинал купюру на два документах с разными номерами и депонирует в зашифрованном виде у банка. Банк выбирает случайным образом и требует раскрыть 99 (или n-1) конвертов, убеждается, что везде написано \$10, а не \$1000, тогда подписывает оставшийся конверт вслепую, не видя номера купюры.

Может быть предусмотрен более простой вариант: за каждым номиналом купюры у банка закреплена своя пара открытых ключей. Тогда подпись присыпается только номер купюры и необходимость проверки номинала перед подписью отпадает.

### Секретное голосование

Слепые подписи используются для секретного голосования. В протоколе Фуджиока, Окамото и Охта, избиратель подготавливает избирательный бюллетень со своим выбором, который он сделал, шифрует его секретным ключом, и маскирует его. Далее избиратель подписывает избирательный бюллетень и посыпает его валидатору. Валидатор проверяет, что подпись принадлежит зарегистрированному избирателю, который еще не голосовал.

Если избирательный бюллетень действителен, валидатор подписывает избирательный бюллетень и возвращает его избирателю. Избиратель удаляет маскировку, раскрывая таким образом зашифрованный избирательный бюллетень, подписанный валидатором. Далее избиратель посыпает в результат полученный подписанный, зашифрованный избирательный бюллетень счётчику. Счётчик проверяет подпись на зашифрованном избирательном бюллетене.

Если избирательный бюллетень действителен, счётчик размещает его в списке, который будет издан после всего голосования. После того, как список издан, избиратели проверяют, что их избирательные бюллетени находятся в списке и посыпают счётчику ключи дешифрования, необходимые, чтобы открыть их избирательные бюллетени. Счётчик использует эти ключи для дешифрования избирательных бюллетеней и добавляет голос к общему числу. После выборов счётчик издает ключи дешифрования наряду с зашифрованными избирательными бюллетенями так, чтобы избиратели могли независимо проверить выбор.

### Полностью слепая подпись

Б - катализ

А - подписать документ, чтобы Б не знал его содержимого.  
Б должен уверить док-т.

1. А берет т и использует маскирующей множитель.

2. А отсылает F(m) Б.

3. Б подписывает и отсылает А.

4. А снимает маскирующей множитель и получает док-т с подписью.

II Протокол работает, если ф-ции подписи и умножения коммутативны.

Вместо сообщения подписывать хэш

$m$ - сообщение

$$h: \{m\} \rightarrow \{0, 1\}^k$$

Формирование подпись:  $s \leftarrow h/m \text{ mod } n$

Проверка подпись:  $h' \leftarrow h/m$

$$\text{Проверка } h'' \leftarrow s \cdot h^d \text{ mod } n$$

// не должно быть коллизий

### Подпись виленскую на основе RSA

1. А выбирает число от 1...n. Затем маскирует  $m$ , получив  $t = m \cdot k^e \text{ mod } n$ .
2. Б подписывает  $t$ :  $t^d = (m \cdot k^e)^d \text{ mod } n$
3. А снимает маскировку с  $t^d$ , получив  $s = t^d / k \text{ mod } n$ .
4. Р-т  $s = \frac{t^d}{k} \text{ mod } n$ .

## 7. Криптосистема Рабина: шифрование с открытым ключом и цифровая подпись

✗ протокол шифрования с открытым ключом. Покаутель зашифрование равен 2, а расшифрование подозрительно.

OK: пара  $(n, B)$ , где  $n = pq$  - составное число,  $B \in \{0, \dots, n-1\}$  - секр., секретное - разложение этого числа на множители.

### Теорема

Задача нахождения всех 4x решений сравнения  $y^2 \equiv x \pmod{n}$  эквивалентна задаче разложения на множители  $n = pq$ .

### Доказательство

$\Rightarrow$  Если известно разложение числа  $n$ , то сравнение может быть решено вычислением квадратных корней из  $d$  по модулям  $p$  и  $q$  и восстановлением рез-та по КТО

Сложность вычислений оценивается по формуле  $\alpha \log n$  не выше степени 3.

$\Leftarrow$  Если известны 4 корня, то они по КТО могут быть представлены в виде:

$$(\sqrt{a} \pmod p, \sqrt{a} \pmod q); (\sqrt{a} \pmod p, -\sqrt{a} \pmod q);$$

$$(-\sqrt{a} \pmod{p}, \sqrt{a} \pmod{q}), (-\sqrt{a} \pmod{p}, -\sqrt{a} \pmod{q}).$$

Если сумма по модулю  $n$  двух корней не равна 0, то она имеет четырехзначный вид с  $n$ , который вычислить двоичным алгоритмом в квадрате с квадратом синтеза от  $\log n$ .

### Схема шифрования Рабина

Вход: ок:  $(n, B)$ , сообщение  $m \in \mathbb{Z}/n\mathbb{Z}$

Выход: число  $p, q$ , шифруется с.

Замыкание:  $c \leftarrow m(m+B) \pmod{n}$

Расшифрование: 1.  $\begin{cases} m_p \equiv \pm \sqrt{c} \pmod{p} \\ m_q \equiv \pm \sqrt{c} \pmod{q} \end{cases}$

получим 4 возможных значения.  
Расшифр-е необходимо.



2. Объединить решения по ктд:

$$m = \sqrt{\frac{B^2}{4} + c} - \frac{B}{2} \pmod{n} = \begin{cases} m_p \pmod{p} \\ m_q \pmod{q} \end{cases}$$

\* От полученных содержит избыточное значение.

Открывший показатель не обратим по модулю  $\varphi(n) \Rightarrow$  Протокол шифрования Рабина по отношению к криптоданнику на основе известных от

Задача расшифрования эквивалентна задаче разложение на множители.

Если разложение числа  $n$  известно, то расшифровать шифр текста можно по определению.

Также возможно расшифровать любое сообщение и это равносильно вычислению квадратного корня. Найти модуль из 4x значений квадр. корня

## Схема подписи Рабина

Вход отправителя. Простые делители  $p$  и  $q$  числа  $n$ .

Вход получателя. Составное число  $n$ .

Для формирования подписи для сообщения  $m$  отправитель выполняет следующие действия.

1. Вырабатывает случайное число  $r$  и вычисляет  $t \equiv h(m||r) \pmod{n}$ , где  $h$  — хэш-функция.

2. Проверяет, является ли  $t$  квадратичным вычетом по модулю  $n$ , для чего вычисляет символы Лежандра  $\left(\frac{t}{p}\right), \left(\frac{t}{q}\right)$ . Если  $t$  — квадратичный невычет по модулю  $n$ , то возвращается на шаг 1 (генерирует новое значение  $r$  и вычисляет новое  $t$ ). В противном случае вычисляет значение  $s \in (\mathbb{Z}/n\mathbb{Z})^*$  такое, что  $s^2 \equiv t \pmod{n}$ , а именно:

- находит  $s_p \leftarrow \sqrt{t} \pmod{p}$ ;
- находит  $s_q \leftarrow \sqrt{t} \pmod{q}$ ;
- восстанавливает  $s$  по китайской теореме об остатках:

$$s \leftarrow s_p q (q^{-1} \pmod{p}) + s_q p (p^{-1} \pmod{q}) \pmod{n}.$$

Подписью для сообщения  $m$  является пара  $(r, s)$ .

Для проверки подписи получатель выполняет следующие действия.

1. Вычисляет значение  $t' \leftarrow s^2 \pmod{n}$  и  $t' \equiv h(m||r)$

2. Если  $t = t'$  то результат: подпись подлинная, иначе результат: подпись неверна. ■

! Подвержена атаке на основе подбора сообщений.

Например: извлечение кв. корня из 4 не требует разложение числа  $n$ .

## 8. Криптосистема Фиата-Шамира: аутентификация и ЦП

### Протокол аутентификации:

1. Доверенный центр генерирует числа  $p \equiv q \equiv 3 \pmod{4}$

2. Вычисляет произведение  $n$

3. Отправляет получившее число всем участникам.

! Разложение числа  $n$  должно быть известно только доверенному центру.

4. Каждый участник выбирает к числу  $s_1, s_2, \dots, s_k \in (\mathbb{Z}/n\mathbb{Z})^*$  и вычисляет  $v_i \equiv s_j^{-2} \pmod{n}$   $\forall j = 1, k$ .

Тогда:  $v_i$  — открытый ключ, набор  $s_i$  — закрытый ключ участника.

Участник A хочет подтвердить личность участнику B.  
Описание и её цикл протокола аутентификации.

A	B
1. $r_i \in (\mathbb{Z}/n\mathbb{Z})^*$ - случайное	1. $\tilde{e} = (e_1, e_2, \dots, e_k), e_j \in \{0,1\}$ .
2. $x_i \equiv r_i^2 \pmod{n}$ .	2. Отправляет $e_i$ участнику A.
3. Отправляет $x_i$ участнику B	3. $z_i \equiv y_i^2 v_1^{e_1} v_2^{e_2} \dots v_k^{e_k} \pmod{n}$ .
4. $y_i \equiv r_i(s_1^A)^{e_1} (s_2^A)^{e_2} \dots (s_k^A)^{e_k} \pmod{n}$	4. Проверяет $z_i \equiv x_i \pmod{n}$ . Если выполняется – переход на следующий шаг.
5. Отправляет $y_i$ участнику B.	

Если равенство выполняется для  $\forall i = 1, 2, \dots, k$ , то участник A проходит аутентификацию.

Общий ОК число  $n = pq$ .  
множ ск:  $S_0, \dots, S_{k-1}$  кольца  $\mathbb{Z}/n\mathbb{Z}$   
множ OK:  $V_0, \dots, V_{k-1}$  кольца  $\mathbb{Z}/n\mathbb{Z}$   
 что  $V_i \in S_i^{-2} \pmod{n}$

Отличие от RSA:  
 использовать общий модуль n,  
 участники не должны  
 знать разложение числа n.

## Схема подписи Фиата-Шамира

**Протокол 4.** Схема подписи Фиата–Шамира.

Вход отправителя. Составное число  $n; s_1, \dots, s_k \in (\mathbb{Z}/n\mathbb{Z})^*$ .

Вход получателя. Составное число  $n; v_1, \dots, v_k \in (\mathbb{Z}/n\mathbb{Z})^*$ , где  $v_i \equiv s_i^{-2} \pmod{n}$ .

Для формирования подписи для сообщения  $m$  отправитель выполняет следующие действия.

- Выбирает произвольное целое число  $r, 1 < r < n - 1$ .
- Полагает  $x \leftarrow r^2 \pmod{n}$ .

3. Вычисляет хэш-функцию от аргумента, представляющего собой конкатенацию чисел  $m$  и  $x$ :  $\tilde{e} \leftarrow h(m||x)$ , и представляет полученное значение в виде двоичного вектора  $(e_1, \dots, e_k)$ .

- Вычисляет  $y \leftarrow r \prod_{i=1}^k s_i^{e_i} \pmod{n}$ .

Подпись для сообщения  $m$  является пара  $(\tilde{e}, y)$ .

Для проверки подписи получатель выполняет следующие действия.

- Представляет число  $\tilde{e}$  в виде двоичного вектора  $(e_1, \dots, e_k)$ .
- Вычисляет  $z \leftarrow y^2 \prod_{i=1}^k v_i^{e_i} \pmod{n}$ .
- Полагает  $e \leftarrow h(m||z)$ .
- Проверяет равенство  $e = \tilde{e}$ . Если оно выполняется, то результат: подпись подлинная, иначе результат: подпись недействительна. ■

Безопасность схемы подписи основана на сложности выделение корня в кольце  $\mathbb{Z}/n\mathbb{Z}$ .

Число  $n$  должно вырабатываться доверенной стороной.

Для того, чтобы убедиться, что число  $n$  выбрано правильно необходимо предусмотреть процедуру доказа-ва числа  $n$ .

# 9. Задача дискретного логарифмирования. Протокол Диффа-Хеллмана. Соотношение между задачей ДХ и ЗДЛ.

**ЗДЛ:** ] Г-группа,  $\langle a \rangle \subseteq G$ ,  $b \in \langle a \rangle$   
Найти  $x \in \mathbb{Z}$ , такое что  $a^x = b$ .

## Протокол Диффа-Хеллмана (уст-ка сеанс. ключа)

Цель: 2 участника могут безопасно обменяться ключом, который в дальнейшем может быть использован.

Причина: только для обмена ключами.

Основан на трудности вычисления дискретных логарифмов.

Безопасность обмена ключами: Много вычислений экспонентов по модулю простого числа  
~~трудно вычислить дискретное логарифм.~~

Для больших простых чисел задача считается неразрешимой.

## Протокол Диффа-Хеллмана

Параметры протокола:  $G$ ,  $\langle a \rangle \subseteq G$

A

B

Возрабатывает случайной  
показатель  $x \in \mathbb{Z}$

воспоследует  $a^x$

отправка  $a^x$

Возрабатывает случайной  
показатель  $y \in \mathbb{Z}$

воспоследует  $a^y$

отправка  $a^y$

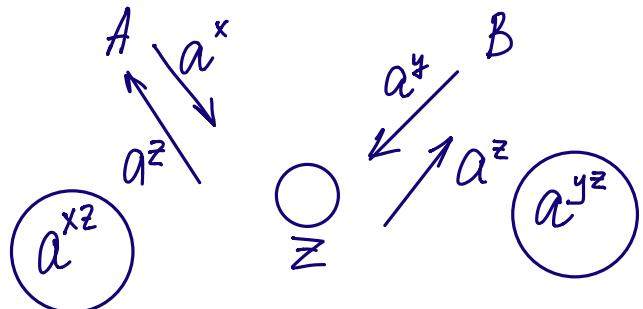
Оба участника возбуждают полученное  
значение в степень, начиная со своим  
показателем.

Оба участника получают ит-е

$a^{xy}$  общий сеансовой ключ

**Задача ДХ:** зная  $a, a^x, a^y$   $\rightarrow a^{xy}$   
(по данному)  $\rightarrow$  (найти)

Криптосистема уязвима к  
атаке "головок посередине".



## 10. Бесключевое шифрование Месси-Омура. Соотношение между задачами Месси-Омура и DH.

Параметры протокола:

группа  $G$ ,  $\langle a \rangle \subseteq G$ ,  $r = \#\langle a \rangle$

внешняя в обе стороны ф-я:  $f: M \rightarrow G$

$r$ - большое простое число  
M-ми-бо сообщ-й

1. Отправитель сообщает  $t \in M$  получает  $t \leftarrow f(m)$ .  
Если  $t$  является единственным элементом группы  $G \Rightarrow$  сообщение не поддается зашифрованию.
2. Отправитель генерирует случайное пок-во  $a \in (\mathbb{Z}/r\mathbb{Z})^*$ ,  
получает элемент  $t_1 \leftarrow t^{a^{-1}}$  группы  $G$  и направляет  $t_1$  получателю.
3. Получатель вырабатывает сл. пок-во  $b \in (\mathbb{Z}/r\mathbb{Z})^*$ ,  
получает  $t_2 \leftarrow (t_1)^b$  и посыпает обратно.
4. Отправитель получает  $t_3 \leftarrow (t_2)^{a^{-1} \pmod r}$  и  
посыпает  $t_3$  получателю.
5. Получатель получает  $t_4 \leftarrow (t_3)^{b^{-1} \pmod r}$ ,  $m \leftarrow f^{-1}(t_4)$

Соотношение между задачами M-O и DH.

Дано:  $\underbrace{t^a}_{a^x}, \underbrace{t^b}_{a^y}, \underbrace{t^{ab}}_a \quad x = \beta^{-1}; y = d^{-1}$

Найти:  $t = \alpha^{xy}$

## II. Протокол Эль-Гамаля шифрования с открытым ключом. Соотношение между ЭГ и DH

В DH оба показателя  $x$  и  $y$  скрытые.

В протоколе Эль-Гамаля один показатель фиксированный и другой секретный ключом расшифр-я, другой - скрытой.

### Протокол Эль-Гамаля

М-ли-во сообщений

$G = \langle a \rangle$  циклическая группа

$f: M \times G \rightarrow G$  (образация по 1-му элементу)

Личный секретный ключом будет пок-ль  $x$ ,

личным открытый ключом значение  $b = a^x$ .

Для расшифр-я сообщение  $m$  отправитель:

1. Генерирует скрытой показателем  $y$ .

2. Вычисляет элементы  $a^y, b^y \in G$  и значение  $c \leftarrow f(m, b^y)$

Шифртекст: пара  $(a^y, c)$

Для расшифрования шифртекста  $(a^y, c)$  получатель:

1. Возбуждай  $a^y$  в степень  $x$ , при этом  $a^{xy} = b^y$

2. Находит сообщение  $m \leftarrow f^{-1}(c, b^y)$  ■

Вид функции  $f$  зависит от группы  $G$ .

Если  $G = F_p^*$  - мультиплик. группа простого поля, то в качестве  $f$  могут использоваться функции:

$$f(m, b^y) = mb^y \pmod{p},$$

$$f(m, b^y) = m + b^y \pmod{p},$$

$$f(m, b^y) = m - b^y \pmod{p}$$

Используя 1 функцию, можно извлечь информацию о сообщении  $m$ , включив символ лемандра:

$$\left(\frac{a}{p}\right), \left(\frac{b}{p}\right), \left(\frac{c}{p}\right) = \left(\frac{m}{p}\right) \left(\frac{b}{p}\right)^y$$

$a$ -кв. неважн

$b$ -кв. может узнать явл. ли  $m$  кв.вал, что соотв. 1 биту инф-ции.

## Соотношение задачи Эль-Гамаля и ДК

Для решения задачи Эль-Гамаля достаточно решить задачу ДК.  
Если для заданных  $a, a^x, a^y$  можно вычислить  $a^{xy}$ , то можно вычислить  $m$  от:  $m = f^{-1}(c, a^{xy})$ .

### 12. Схема ЦП Эль-Гамаля. Задачи, положенные в основу безопасности.

Параметры протокола:  $a$  - образующая подгруппы простого порядка  $r$  мультипликативной группы  $\mathbb{F}_p^*$ .  
 $X$  - квадрат подпись (число  $0 < X < r$ )  
 $b = a^X \pmod p$  - квадрат проверки

#### Формирование подписи:

$m$  - сообщение,  $0 < m < r$

Отправитель: 1. Генерирует сл. число  $k$  ( $0 < k < r$ )

$$2. w \leftarrow a^k \pmod p$$

$$3. \text{Находит число } s \leftarrow (m - Xw)k^{-1} \pmod r$$

Подпись для сообщения  $m$  будет пара  $(w, s)$ .

#### Проверка подписи:

1. Проверка неравенства  $w < p$

если не выполняется, то подпись недействительна.

2. Проверка сравнений:  $a^m \equiv b^w w^s \pmod p$

если выполняется, то подпись подлинная.

Проверка  $w < p$  необходима, иначе из подлинной подписи  $(w, s)$  можно сформировать подпись для произвольного сообщения  $m'$ , не имея квадрат подписи.

Можно вычислить значение  $\beta \equiv m' m^{-1} \pmod r$ .

$$\text{Тогда } a^{m'} \equiv a^{\beta m} \equiv b^{\beta w} w^{\beta s} \pmod p.$$

Полагая, что  $s' \equiv \beta s \pmod r$ ,  $w' \equiv \beta w \pmod r$ ,  $n' \equiv w \pmod p$  можно найти целое  $w' < pr$ , по КТД, при этом  $(n', s')$  будет корректной подписью для сообщения  $m'$ .

# 13. Эллиптическая кривая. Групповой закон. Число точек. Проективная плоскость. Понятие бесконечно удаленной точки.

## Общие сведения:

Группа — непустое множество  $G$  с бинарной операцией  $*$ , обычно называемой умножением, для которой выполняются следующие условия.

1. Замкнутость: для любых  $a, b \in G$  выполняется  $a * b \in G$ .
2. Ассоциативность: для любых  $a, b, c \in G$  выполняется  $(a * b) * c = a * (b * c)$ .
3. В множестве  $G$  существует единичный элемент  $e$ , такой, что  $c * e = e * c = c$  для любого  $c \in G$ .
4. Для любого  $a \in G$  существует обратный элемент  $a^{-1} \in G$ , такой, что  $a^{-1} * a = a * a^{-1} = e$ .

Группа  $G$  называется абелевой, или коммутативной, если  $a * b = b * a$  для любых элементов  $a, b \in G$ .

Если в группе конечное число элементов, то группа называется конечной, а число её элементов называется порядком группы.

Подгруппа — подмножество группы, которое само является группой.

Циклическая группа — группа, в которой существует такой элемент  $a$ , что любой другой элемент  $b$  этой группы можно представить в виде  $b = a^k$  для некоторого целого числа  $k$ . Элемент  $a$  называется образующей группы. Циклическая группа, образованная элементом  $a$ , обозначается  $\langle a \rangle$ .

Поле — множество  $F$ , на котором заданы две операции — сложение  $+$  и умножение  $*$ , — состоящее не менее чем из двух элементов — нулевого  $0$  и единичного  $1$ . При этом выполняются следующие условия.

1. Относительно операции сложения множество  $F$  является абелевой группой с нулевым элементом.
2. Относительно операции умножения множество  $F$  является абелевой группой с единичным элементом  $1$ .
3. Операция умножения дистрибутивна относительно операции сложения, то есть для любых элементов  $a, b, c \in F$  выполняется равенство  $(a + b) * c = a * c + b * c$  и  $a * (b + c) = a * b + a * c$ .

Поле из  $p$  элементов, где  $p$  — простое число, обозначается  $F_p$ .

Эллиптическая кривая  $E$  над полем  $F_p$  — множество точек  $(x, y)$ , где  $x, y \in F_p$ , удовлетворяющих уравнению  $y^2 \equiv x^3 + ax + b \pmod{p}$ , и бесконечно удаленная точка  $P_\infty$ , явн. нулевым элементом группой.

Закон сложения точек ЭК:

$$P_1 = (x_1, y_1)$$

$$1) -P_1 = (x_1, -y_1)$$

$$P_2 = (x_2, y_2)$$

$$2) \text{ если } P_1 = P_2, \text{ то}$$

$$P_3 = P_1 + P_2 = (x_3, y_3)$$

$$x_3 \equiv ((3x_1^2 + a) \cdot (2y_1)^{-1})^2 - 2x_1 \pmod{p}$$

$$y_3 \equiv -y_1 + (3x_1^2 + a) \cdot (2y_1)^{-1} (x_1 - x_3) \pmod{p}$$

$$3) \text{ если } P_1 \neq P_2, \text{ то}$$

$$x_3 \equiv ((y_2 - y_1) \cdot (x_2 - x_1)^{-1})^2 - x_1 - x_2 \pmod{p},$$

$$y_3 \equiv -y_1 + (y_2 - y_1) \cdot (x_2 - x_1)^{-1} \cdot (x_1 - x_3) \pmod{p}$$

Относительно указанной операции сложения мн-во точек ЭК является абелевой группой.

# 14. ЦП по стандарту ГОСТ Р 34.10-2012

Стандарт использует ЭК над конечными полями.

## Параметры схемы ЦП

1)  $p$  - простое число  
 $p$  - модуль ЭК, удовлетворяющий неравенству  $p > 2^{255}$

2) ЭК  $E$ :  $y^2 \equiv x^3 + ax + b \pmod{p}$  задается инвариантом

$$J(E) = 1728 \cdot 4a^3 \cdot (4a^3 + 27b^2)^{-1} \pmod{p}$$

или коэффициентами,  $a, b \in F_p$ , где  $4a^3 + 27b^2 \neq 0 \pmod{p}$   
 $a \equiv 3J(E) \cdot (1728 - J(E))^{-1} \pmod{p}$   
 $b \equiv 2J(E) \cdot (1728 - J(E))^{-1} \pmod{p}$

3)  $m \in \mathbb{Z}$  порядок циклической подгруппы группы точек ЭК  $E$ ,  $|p+1-m| \leq 2\sqrt{p}$

4)  $q$  - простое порядок циклической подгруппы группы точек ЭК  $E$   
из  $m = nq$ ,  $n \geq 1$  - целое число  
 $2^{254} < q < 2^{256}$  или  $2^{508} < q < 2^{512}$

5) точка  $P \neq O$  ЭК  $E$ ,  $P = (x_p, y_p)$ , удовлетв. рав-ву  $qP = O$  -  
обращающая циклическую подгруппу порядка  $q$ .

6) хэм-функция  $h: V^* \rightarrow V_e$  ( $h: f_0, f_1 \mapsto f_0, f_1$ )<sup>отобр.</sup>  
сообщения, представляемые в виде двоичных векторов произ-  
вольной конечной длины, в двоичные векторы длины  
 $\ell$  бит.

Хэм-ф-ция определена в ГОСТ Р 34.11-2018.

## Требования к параметрам схемы ЦП

- должно быть выполнено условие  $p^t \neq 1 \pmod{q}$  для всех целых  $t = 1, 2, \dots, B$ , где  $B = 31$ , если  $2^{254} < q < 2^{256}$ , и  $B = 131$ , если  $2^{508} < q < 2^{512}$ ;

- должно быть выполнено неравенство  $m \neq p$ ;

- инвариант кривой должен удовлетворять условию  $J(E) \neq 0 \pmod{p}$  и  $J(E) \neq 1728 \pmod{p}$  (эквивалентное условие:  $a \neq 0 \pmod{p}$  и  $b \neq 0 \pmod{p}$ ).

## Личные ключи

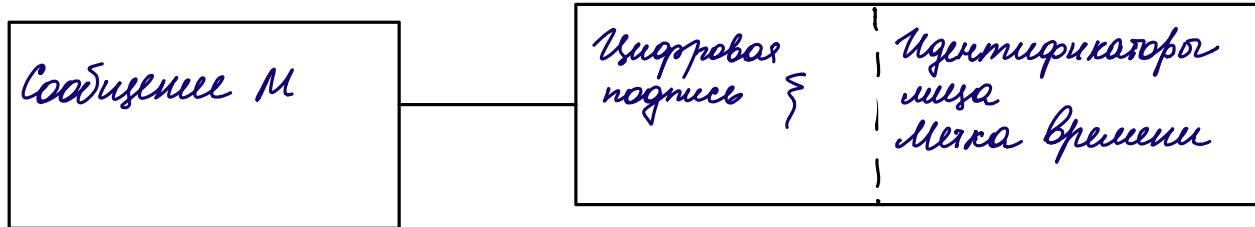
Ключ подписи: целое число  $d$ , удовлетворяющее  $0 < d < q$

Ключ проверки подписи: точка  $Q = (x_Q, y_Q)$ , такая, что  
 $Q = dP = \underbrace{P + \dots + P}_{d \text{ раз}}$

# Схема подписанного сообщения.

## Назначение:

- ✓ аутентификация лица, подписавшего электр. сообщение
- ✓ возможность обеспечить контроль целостности сообщения
- ✓ док-во подтверждать авторство лица подписавш. сообщ-ия
- ✓ защитить сообщение от возможной подделки



# Формирование подписи

## 6. Формирование подписи

Исходные данные: сообщение  $M \in V^*$ , ключ подписи  $d$ .

**Шаг 1.** Вычислить хэш-код сообщения  $M$ :  $\bar{h} = h(M)$ .

**Шаг 2.** Вычислить целое число  $\alpha$ , двоичным представлением которого является вектор  $\bar{h}$ , и определить  $e \equiv \alpha \pmod q$ .

Если  $e = 0$ , то положить  $e = 1$ .

**Шаг 3.** Сгенерировать случайное (псевдослучайное) целое число  $k$ , удовлетворяющее неравенству  $0 < k < q$ .

**Шаг 4.** Вычислить точку эллиптической кривой:

$C = kP, C = (x_C, y_C)$ , и определить  $r \equiv x_C \pmod q$ .

Если  $r = 0$ , то вернуться к шагу 3.

**Шаг 5.** Вычислить значение  $s \equiv (r \cdot d + k \cdot e) \pmod q$ .

Если  $s = 0$ , то вернуться к шагу 3.

**Шаг 6.** Вычислить двоичные векторы  $\bar{r}$  и  $\bar{s}$ , соответствующие числам  $r$  и  $s$ , и определить цифровую подпись  $\zeta = (\bar{r} \parallel \bar{s})$  как конкатенацию двух двоичных векторов.

**Выходной результат:** цифровая подпись  $\zeta$ .

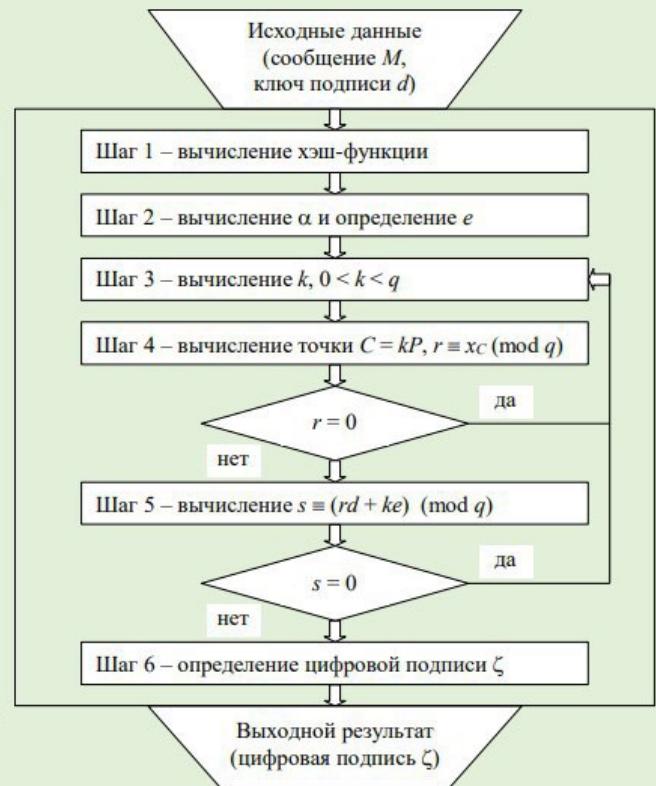


Рисунок 2 – Схема процесса формирования цифровой подписи

# Проверка цифровой подписи

**Исходные данные:** подписанное сообщение  $M$ , цифровая подпись  $\zeta$ , ключ проверки  $Q$ .

**Шаг 1.** По полученной подписи  $\zeta$  вычислить целые числа  $r$  и  $s$ . Если выполнены неравенства  $0 < r < q$ ,  $0 < s < q$ , то перейти к следующему шагу.

В противном случае **подпись неверна**.

**Шаг 2.** Вычислить хэш-код полученного сообщения  $M$ :  $\bar{h} = h(M)$ .

**Шаг 3.** Вычислить целое число  $\alpha$ , двоичным представлением которого является вектор  $\bar{h}$ , и определить  $e \equiv \alpha \pmod{q}$ . Если  $e = 0$ , то положить  $e = 1$ .

**Шаг 4.** Вычислить значение  $v \equiv e^{-1} \pmod{q}$ .

**Шаг 5.** Вычислить значения

$$z_1 \equiv s \cdot v \pmod{q}, z_2 \equiv -r \cdot v \pmod{q}.$$

**Шаг 6.** Вычислить точку эллиптической кривой:  $C = z_1P + z_2Q$ ,  $C = (x_C, y_C)$ , и определить  $R \equiv x_C \pmod{q}$ .

**Шаг 7.** Если выполнено равенство  $R = r$ , то **подпись принимается**, в противном случае **подпись неверна**.

**Выходной результат:** свидетельство о достоверности или ошибочности цифровой подписи  $\zeta$ .

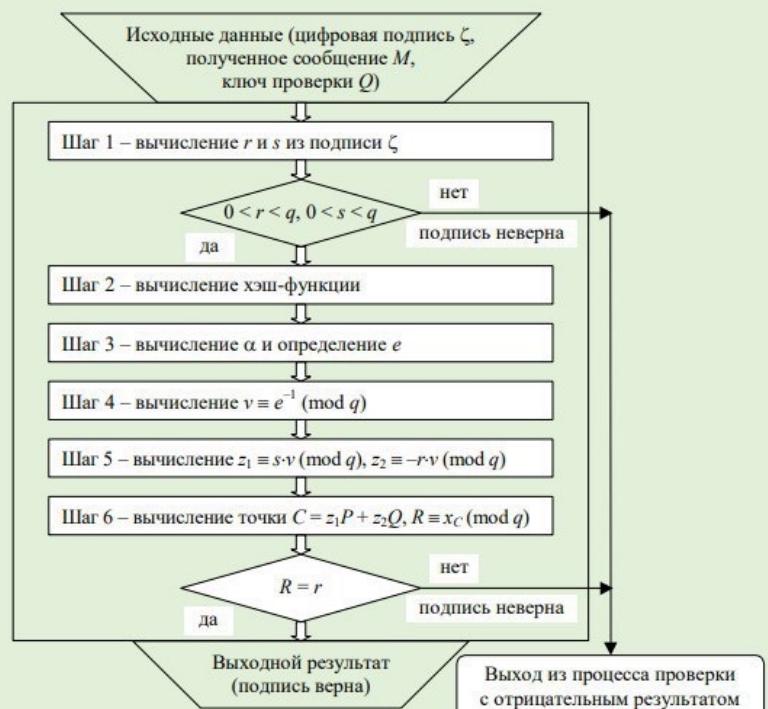


Рисунок 3 – Схема процесса проверки цифровой подписи

# 15. Методы решения задачи дискретного логарифмирования на ЭК: метод Полига-Хеллмана. Метод Полларда, параллельный метод Полларда

## Задача ДЛ в группе точек ЭК

Дано: точка  $P$  ЭК - образующая цикл. группы  $\langle P \rangle$  простого порядка  $q$ ;

точка  $Q$  ЭК - элемент цикл. группы  $\langle P \rangle$ .

Найти: целое  $d$ ,  $0 < d < q$ , что  $Q = dP$

## ЗДЛ, лежащие в основе безопасности ГОСТ Р 34.10-2018

1. По образующей  $P$  и кюлю проверки подписи  $Q$  найти кюль подписи  $d$ .

2. По образующей  $P$  и точке  $C$  найти рабочий кюль  $k$ .

Из части подписи  $r$  восстановить координату  $x_c$  точки  $C$ .  
По координате  $x_c$  восстановить  $y_c$ :  $y^2 \equiv x_c^3 + ax_c + b \pmod{p}$

Снять кюли  $k$  и  $d$ :  $d \equiv (s - k \cdot e) \cdot r^{-1} \pmod{q}$

### 3.1. Метод Полига-Хеллмана

В основе метода: основная теорема об абелевых группах (конечная абелева группа раскладывается в прямое произведение циклических групп, порядки которых являются степенями простых чисел).

Пусть  $\#\langle P \rangle = q = \prod_{j=1}^t q_j^{\alpha_j}$ . Необходимо вычислить  $d_j \equiv d \pmod{q_j^{\alpha_j}}$ ,  $j = 1, \dots, t$ .

Для  $j = 1, \dots, t$  число  $d_j$  представляем в виде

$$d_j \equiv z_0 + z_1 q_j + z_2 q_j^2 + \dots + z_{\alpha_j-1} q_j^{\alpha_j-1} \pmod{q_j^{\alpha_j}}, \quad 0 \leq z_i \leq q_j - 1.$$

Пусть  $P_0 = \frac{q}{q_j} P$ , тогда  $\#\langle P_0 \rangle = q_j$ . Рассмотрим

$$Q_0 = \frac{q}{q_j} Q = \frac{q}{q_j} (d_j P) = \frac{q}{q_j} (z_0 + z_1 q_j + z_2 q_j^2 + \dots) P = z_0 \left( \frac{q}{q_j} P \right) + z_1 \underbrace{q_j P}_{P_\infty} + z_2 q_j \underbrace{q_j P}_{P_\infty} + \dots = z_0 P_0.$$

Решаем задачу дискретного логарифмирования:

Дано:

точка  $P_0$  эллиптической кривой — образующая циклической группы  $\langle P_0 \rangle$  малого простого порядка  $q_j$ ;

точка  $Q_0$  эллиптической кривой — элемент циклической группы  $\langle P P_0 \rangle$ .

Найти:

целое число  $z_0$ ,  $0 < d < q_j$ , такое что  $Q = z_0 P$ .

Далее

$$Q_1 = \frac{q}{q_j^2} \left( \frac{Q}{d_j P} - z_0 P \right) = (d_j - z_0) \left( \frac{q}{q_j^2} P \right) = (z_1 q_j + z_2 q_j^2 + \dots) \left( \frac{q}{q_j^2} P \right) = z_1 \left( \frac{q}{q_j} P \right) + z_2 \underbrace{\left( \frac{q}{q_j} P \right)}_{P_\infty} + \dots = z_1 P_0.$$

Тогда  $z_1 = \log_{P_0} Q_1$ .

На каждой итерации находим  $z_k = \log_{P_0} Q_k$ , вычисляя точку

$$Q_k = \frac{q}{q_j^{k+1}} (Q - z_0 P - z_1 q_j P - \dots - z_{k-1} q_j^{k-1} P), k = 1, \dots, \alpha_j - 1.$$

По китайской теореме об остатках восстанавливаем значение  $d$ :

$$\begin{cases} d \equiv d_1 \pmod{q_1^{\alpha_1}}, \\ d \equiv d_2 \pmod{q_2^{\alpha_2}}, \\ \dots \\ d \equiv d_t \pmod{q_t^{\alpha_t}}. \end{cases}$$

### 3.2. Метод Полларда

Требует использования «случайного» отображения  $f$ , обладающего сжимающими свойствами и вычислимостью логарифма (логарифм точки  $f(T)$  можно выразить через  $d$  и логарифм точки  $T$ ).

Основная цель метода: поиск пар  $(\alpha', \beta')$  и  $(\alpha'', \beta'')$ ,

$$\alpha'P + \beta'Q = \alpha''P + \beta''Q.$$

Тогда  $(\alpha' - \alpha'')P = (\beta'' - \beta')Q = (\beta'' - \beta')dP$ , и

$$d \equiv (\alpha' - \alpha'')(\beta'' - \beta')^{-1} \pmod{q}.$$

Используются ветвящиеся отображения  $f: \langle P \rangle \rightarrow \langle P \rangle$  вида

$$f(T) = T + a_j P + b_j Q, j = H(T).$$

Пусть  $\langle P \rangle = S_1 \cup S_2 \cup \dots \cup S_L$ , где  $L = 16$  или  $L = 32$ . Например, если  $L = 32$ , то  $T \in S_j$ , если  $x_T \equiv j \pmod{32}$ .

Значение функции ветвлений  $H: \langle P \rangle \rightarrow \{1, 2, \dots, L\}$  равно  $H(T) = j$ .

Коэффициенты  $a_j, b_j \in \mathbb{Z}/q\mathbb{Z}$ ,  $j = \{1, \dots, 32\}$ .

Вычислимость логарифма. Если  $T = \alpha P + \beta Q$ , то

$$f(T) = T + a_j P + b_j Q = (\alpha + a_j)P + (\beta + b_j)Q = \alpha'P + \beta'Q.$$

### 3.3. Параллельный метод Полларда

Каждый вычислитель — одно и то же отображение, но с разными начальными точками, тогда одинаковые точки могут встретиться не в цикле («голова» буквы  $\rho$ ), а на дереве («хвост» буквы  $\rho$ ):

**Пример 3.** Пусть  $E(\mathbf{F}_{73})$ :  $y^2 \equiv x^3 + 3x + 9$ ,  $q = 83$ ,  $P = (31, 6)$ ,  $Q = (24, 53)$ .

Отображение

$$f((x, y)) = \begin{cases} (x, y) + P, & \text{если } x \text{ нечетное,} \\ (x, y) + Q, & \text{если } x \text{ четное.} \end{cases}$$

	$i$	0	1	2	3	4
1-й вычислитель	$Q_i$	$P = (31, 6)$	$(17, 3)$	$(50, 16)$	$(48, 71)$	$(70, 22)$
	$\log(Q_i)$	1	2	3	$d + 3$	$2d + 3$
2-й вычислитель	$Q_i$	$Q = (24, 53)$	$(61, 17)$	$(56, 70)$	$(28, 27)$	$(45, 47)$
	$\log(Q_i)$	$d$	$2d$	$2d + 1$	$3d + 1$	$4d + 1$
3-й вычислитель	$Q_i$	$5P = (63, 1)$	$(28, 46)$	$(56, 3)$	$(2, 13)$	$(31, 67)$
	$\log(Q_i)$	5	6	$d + 6$	$2d + 6$	$3d + 6$

## 16. Метод решения ЗДЛ на ЭК: метод Полларда в группе автоморфизмов.

### 3.4. Метод Полларда в группе автоморфизмов

На эллиптической кривой может существовать автоморфизм  $\varphi: \langle P \rangle \rightarrow \langle P \rangle$ , который вычисляется быстрее, чем умножение точки на число.

Пусть  $t = \text{ord } \varphi$ :

$$\varphi^t(R) = R$$

для любой точки  $R \in \langle P \rangle$ .

Отношение эквивалентности:  $R_1 \sim R_2 \Leftrightarrow \exists j, 0 \leq j \leq t-1: R_1 = \varphi^j(R_2)$ .

Класс эквивалентности:  $[R] = \{R, \varphi(R), \varphi^2(R), \dots, \varphi^{t_1-1}(R)\}$ , где  $t_1 | t$ .

$\overline{R}$  — канонический представитель класса  $[R]$ .

Тогда ветвящееся отображение:  $g(R) = \overline{f(R)}$ .

*Вычислимость логарифма.*

Пусть  $\varphi(P) = \lambda P$ , где  $0 \leq \lambda \leq q$ , где  $q = \#\langle P \rangle$ . Тогда  $\varphi(R) = \lambda R$  для любой точки  $R \in \langle P \rangle$ .

Если  $R = \alpha P + \beta Q$ , то  $\overline{R} = \varphi^j(R) = \varphi^j(\alpha P + \beta Q) = (\lambda^j \alpha \bmod q)P + (\lambda^j \beta \bmod q)Q$ .

Список приводит классификацию:

- ✓ Широкополосные (число бит в скрят. сообщении сравнило с числом бит в секр. числе)
- ✓ УзкоПолосные (количество бит значи-ко меньше кол-ва бит в секр. числе.)



# 17. Понятие скрытого канала. Скрытый канал в схеме цп Эль Гамаля.

## Устранение скрытого канала.

Схема ЦП имеет особенность, позволяющую подписывающему спрятать в подписи инф-цию, которая может быть вывлечена при знании доп.секрета. Подписи не отличаются от обычных.

В скрытом канале помимо подписи зависит от "сугайно" числа. Перебирая разнг. кандидатов, подпись может управлять видом подписи и закодировавшее инф-цию.

### Формирование подписи

**Исходные данные:** сообщение  $M \in V^*$ , ключ подписи  $d$ .

**Шаг 1.** Вычислить хэш-код сообщения  $M: \bar{h} = h(M)$ .

**Шаг 2.** Вычислить целое число  $\alpha$ , двоичным представлением которого является вектор  $\bar{h}$ , и определить  $e \equiv \alpha \pmod{q}$ .

Если  $e = 0$ , то положить  $e = 1$ .

**Шаг 3.** Сгенерировать случайное (псевдослучайное) целое число  $k$ , удовлетворяющее неравенству  $0 < k < q$ .

**Шаг 4.** Вычислить точку эллиптической кривой:

$C = kP, C = (x_C, y_C)$ , и определить  $r \equiv x_C \pmod{q}$ .

Если  $r = 0$ , то вернуться к шагу 3.

**Шаг 5.** Вычислить значение  $s \equiv (r \cdot d + k \cdot e) \pmod{q}$ .

Если  $s = 0$ , то вернуться к шагу 3.

**Шаг 6.** Вычислить двоичные векторы  $\bar{r}$  и  $\bar{s}$ , соответствующие числам  $r$  и  $s$ , и определить цифровую подпись  $\zeta = (\bar{r} \parallel \bar{s})$  как конкатенацию двух двоичных векторов.

**Выходной результат:** цифровая подпись  $\zeta$ .

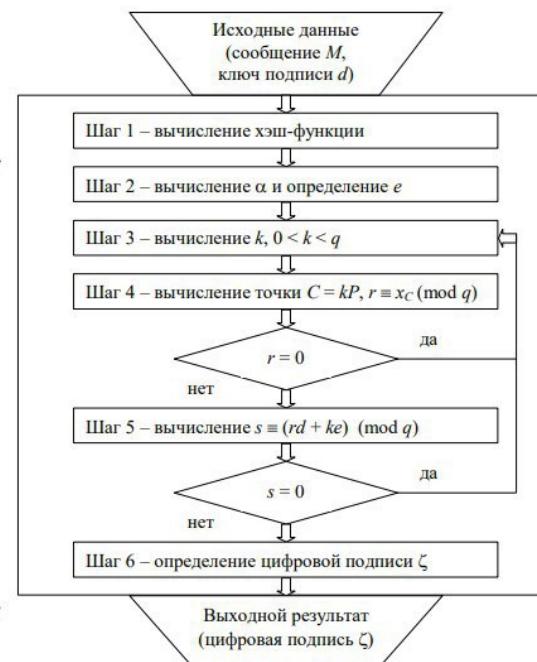


Рисунок 2 – Схема процесса формирования цифровой подписи

### Проверка цифровой подписи

**Исходные данные:** подписанное сообщение  $M$ , цифровая подпись  $\zeta$ , ключ проверки  $Q$ .

**Шаг 1.** По полученной подписи  $\zeta$  вычислить целые числа  $r$  и  $s$ . Если выполнены неравенства

$0 < r < q, 0 < s < q$ , то перейти к следующему шагу.

В противном случае **подпись неверна**.

**Шаг 2.** Вычислить хэш-код полученного сообщения  $M: \bar{h} = h(M)$ .

**Шаг 3.** Вычислить целое число  $\alpha$ , двоичным представлением которого является вектор  $\bar{h}$ , и определить  $e \equiv \alpha \pmod{q}$ . Если  $e = 0$ , то положить  $e = 1$ .

**Шаг 4.** Вычислить значение  $v \equiv e^{-1} \pmod{q}$ .

**Шаг 5.** Вычислить значения  $z_1 \equiv s \cdot v \pmod{q}, z_2 \equiv -r \cdot v \pmod{q}$ .

**Шаг 6.** Вычислить точку эллиптической кривой:  $C = z_1P + z_2Q, C = (x_C, y_C)$ , и определить  $R \equiv x_C \pmod{q}$ .

**Шаг 7.** Если выполнено равенство  $R = r$ , то **подпись принимается**, в противном случае **подпись неверна**.

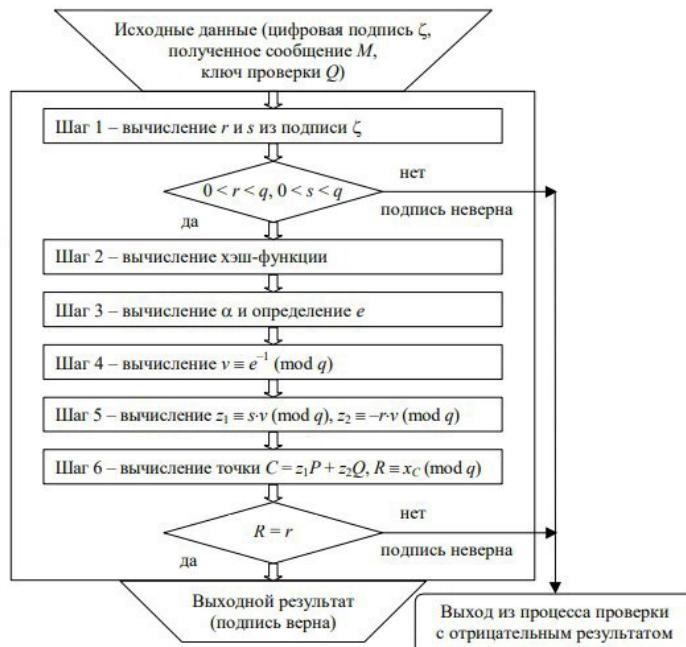


Рисунок 3 – Схема процесса проверки цифровой подписи

**Выходной результат:** свидетельство о достоверности или ошибочности цифровой подписи  $\zeta$ .

## Устранение скрытого канала

1. Отправитель генерирует случайное целое число  $k$ ,  $0 < k < q$ , вычисляет точку  $C_1 \leftarrow kP$  и направляет ее контролеру.
2. Контролер запоминает точку  $C_1$ , генерирует случайное целое число  $k'$ ,  $0 < k' < q$ , и возвращает его отправителю.
3. Отправитель вычисляет показатель  $k_1 \leftarrow kk' \pmod{q}$ , точку  $C_2 \leftarrow k_1C_1$ ,  $C_2 = (x_2, y_2)$ , находит подпись  $s \leftarrow k_1e + dx_2 \pmod{q}$  и передает сообщение  $M$  с подписью  $(x_2, s)$  контролеру.
4. Контролер проверяет, что подпись для сообщения  $M$  правильная и что  $x_2$  — это  $x$ -координата точки  $k'C_1$ , и в случае успешной проверки передает подписанное сообщение  $(M, x_2, s)$  получателю.

## 18. Схема подписи вслепую на основе Эль Гамаля

Уравнение формирования подписи:  $s = f(C) \cdot d + k \cdot m$

Подписью является пара  $(C, s)$ .

Проверка подписи:  $sP = f(C)Q + mC$

Формирование другого правильно подписанного сообщения.

1. Выбрать произвольный показатель  $\alpha$ , положить  $k' = \alpha k$  (значение  $k$  неизвестно, поэтому  $k'$  тоже неизвестно). Этому показателю будет соответствовать точка  $C' = \alpha k P = \alpha C$ .
2. Вычислить показатель  $\beta \equiv f(C')(f(C))^{-1} \pmod{q}$ .
3. Вычислить новое сообщение и подпись  $m' = \alpha^{-1} \beta m, s' = \beta s$ .

P.S. есть пример про банк и кивиита.

## 19. Проверка на простоту с использованием ЭК

**Утверждение.** Существует вероятностный алгоритм доказательства простоты. При этом для любого натурального числа  $k$  доля  $k$ -значных простых чисел, для которых среднее время работы алгоритма полиномиально, не меньше, чем

$$1 - O(2^{-k^{\frac{c}{\log \log k}}}).$$

**Алгоритм.**

1. Положить  $p_0 \leftarrow n, i \leftarrow 0$ . Выбрать такое натуральное  $k$ , что  $2^{k-1} < p_0 < 2^k$ .
2. Выбрать случайные коэффициенты  $A, B \in \mathbb{Z}/p_i\mathbb{Z}$  эллиптической кривой и проверить, что  $D = \text{НОД}(4A^3 + 27B^2, p_i) = 1$ .
  - 2.1. Если  $i = 0$  и  $1 < D < p_0$ , то результат: число  $p_0 = n$  — составное.
  - 2.2. Если  $i > 0$  и  $1 < D < p_i$ , то вернуться на шаг 1.
  - 2.3. Если  $i > 0$  и  $D = p_i$ , то вернуться к началу шага 2 и выбрать другие  $A$  и  $B$ .

3. В предположении, что  $p_i$  — простое число, найти число точек  $\#E(\mathbb{Z}/p_i\mathbb{Z})$  кривой  $y^2 \equiv x^3 + Ax + B \pmod{p_i}$ . Если найденное число  $\#E(\mathbb{Z}/p_i\mathbb{Z})$  нечётно, то вернуться на шаг 2. Иначе положить  $q = \frac{\#E(\mathbb{Z}/p_i\mathbb{Z})}{2}$  и проверить выполнение неравенства (теорема Хассе):  $|2q - (p_i + 1)| < 2\sqrt{p_i}$ .

3.1. Если это неравенство не выполняется при  $i > 0$ , то вернуться на шаг 1.

3.2. Если оно не выполняется при  $i = 0$ , то **результат**: число  $p_0 = n$  — составное.

4. Проверить число  $q$  на простоту алгоритмом Миллера–Рабина (или Соловэя–Штрассена). Если число  $q$  составное, то вернуться на шаг 2.

5. Выбрать случайную точку  $P \in E(\mathbb{Z}/p_i\mathbb{Z})$  и проверить, что  $2qP = P_\infty$ .

5.1. Если это неравенство не выполняется при  $i > 0$ , то вернуться на шаг 1.

5.2. Если оно не выполняется при  $i = 0$ , то **результат**: число  $p_0 = n$  — составное.

6. Положить  $p_{i+1} \leftarrow q$ .

7. Проверить, что  $q \leq 2^{k^{\frac{c}{\log \log k}}}$  для некоторой константы  $c$  (константа обусловлена дополнительным алгоритмом проверки чисел на простоту).

7.1. Если неравенство не выполнено, то  $i \leftarrow i + 1$  и вернуться на шаг 2.

7.2. Если неравенство выполнено, то дополнительно проверить число  $q$  на простоту.

7.2.1. Если  $q$  оказалось составным, то вернуться на шаг 1.

7.2.2. Если  $q$  простое, то **результат**: число  $n$  — простое. □

**Теорема (Поклингтон).** Пусть  $n \in \mathbb{Z}, n \geq 3, n = QR + 1$ , где  $\text{НОД}(Q, R) = 1, R < Q$  и  $Q = \prod_{j=1}^t q_j^{\alpha_j}$ . Если для каждого  $\forall q_j \exists a_j \in \mathbb{Z}$ , для которого  $a_j^{n-1} \equiv 1 \pmod{n}$  и  $\text{НОД}(a_j^{\frac{n-1}{q_j}} - 1, n) = 1$ , то число  $n$  простое.

**Теорема.** Пусть  $n \in \mathbb{N}, n > 1, E(\mathbb{Z}/n\mathbb{Z})$  — кривая, определяемая уравнением  $y^2 \equiv x^3 + Ax + B \pmod{n}$ ,  $\text{НОД}(4A^3 + 27B^2, n) = 1$ . Пусть  $m \in \mathbb{Z}, q|m$  — простое число,  $q > (\sqrt[4]{n} + 1)^2$ . Если существует такая точка  $P \in E(\mathbb{Z}/n\mathbb{Z})$ , что 1)  $mP = P_\infty$ ; 2) точка  $\frac{m}{q}P$  определена и не равна  $P_\infty$ , то  $n$  — простое число.

**Доказательство.** Пусть  $n$  составное, и  $p \leq \sqrt{n}, p|n$ . Пусть  $m'$  — порядок группы точек кривой  $E(\mathbb{Z}/p\mathbb{Z})$ , заданной тем же уравнением, что и  $E(\mathbb{Z}/n\mathbb{Z})$ . По теореме Хассе

$$m' \leq p + 1 + 2\sqrt{p} = (\sqrt{p} + 1)^2 \leq (\sqrt[4]{n} + 1)^2 < q,$$

значит,  $\text{НОД}(q, m') = 1$  и существуют такие  $u, v \in \mathbb{Z}$ , что  $uq + vm' = 1$ , то есть  $uq \equiv 1 \pmod{m'}$ .

Пусть  $P' \equiv P \pmod{p}$ . Тогда на кривой  $E(\mathbb{Z}/p\mathbb{Z})$  имеем:  $\frac{m}{q}P' = 1 \cdot \frac{m}{q}P' = uq \cdot \frac{m}{q}P' = umP' = P_\infty$ . Но при этом из п. 2:  $\frac{m}{q}P' \equiv \frac{m}{q}P \neq P_\infty$ . Противоречие. □

В алгоритме в случае успеха строится цепочка  $n = p_0 > p_1 > \dots > p_l$ , из простоты  $p_l$  следует простота числа  $n$ . Эта цепочка является *сертификатом* простоты, с его помощью вторичная проверка данного  $k$ -значного числа может быть выполнена с полиномиальной сложностью.

Недостаток — многократное вычисление числа точек. Можно использовать эллиптические кривые с комплексным умножением, для которых порядок группы определяется легко.

## 20. Разложение числа на множители с использованием ЭК

Метод предложен в 1987 г. Автор: Hendrik Lenstra.

Является аналогом  $(p - 1)$ -метода Полларда:

1. Выбрать базу разложения  $B = \{p_1, p_2, \dots, p_s\}$ .
2. Выбрать случайное целое  $a$ ,  $2 \leq a \leq n - 2$ , и вычислить  $d \leftarrow \text{НОД}(a, n)$ . При  $d \geq 2$  положить  $p \leftarrow d$  и результат:  $p$ .
3. Для  $i = 1, 2, \dots, s$  выполнить следующие действия.
  - 3.1. Вычислить  $l \leftarrow \left\lceil \frac{\ln n}{\ln p_i} \right\rceil$ .
  - 3.2. Положить  $a \leftarrow a^{p_i^l} \pmod{n}$ .
  - 3.3. Вычислить  $d \leftarrow \text{НОД}(a - 1, n)$ .
  - 3.4. При  $d = 1$  или  $d = n$  результат: «Делитель не найден». В противном случае положить  $p \leftarrow d$  и результат:  $p$ .

Если  $n$  раскладывается на множители:  $n = pq$ ,  $p \neq q$  – простые числа, то по к.т.о.:

$$E(\mathbf{Z}/n\mathbf{Z}) \cong E(\mathbf{Z}/p\mathbf{Z}) \oplus E(\mathbf{Z}/q\mathbf{Z}).$$

Умножение любого элемента группы  $E(\mathbf{Z}/n\mathbf{Z})$  на  $m \in \mathbf{Z}$  сохраняет данный изоморфизм:

$$mE(\mathbf{Z}/n\mathbf{Z}) \cong mE(\mathbf{Z}/p\mathbf{Z}) \oplus mE(\mathbf{Z}/q\mathbf{Z}).$$

Закон сложения:

если  $P_1 = P_2$ , то  $x_3 = ((3x_1^2 + a) \cdot (2y_1)^{-1})^2 - 2x_1$ ,  $y_3 = -y_1 + (3x_1^2 + a) \cdot (2y_1)^{-1} \cdot (x_1 - x_3)$ ;

если  $P_1 \neq P_2$ , то  $x_3 = ((y_2 - y_1) \cdot (x_2 - x_1)^{-1})^2 - x_1 - x_2$ ,  $y_3 = -y_1 + (y_2 - y_1) \cdot (x_2 - x_1)^{-1} \cdot (x_1 - x_3)$ .

**Теорема.** Пусть  $E(\mathbf{Z}/n\mathbf{Z})$  задана уравнением  $y^2 = x^3 + Ax + B$ ,  $\text{НОД}(4A^3 + 27B^2, n) = 1$  (то есть полином  $x^3 + Ax + B$  не имеет кратных корней). Пусть  $P_1$  и  $P_2$  – точки кривой  $E(\mathbf{Z}/n\mathbf{Z})$ , знаменатели координат которых взаимно просты с  $n$ ,  $P_1 \neq -P_2$ . Знаменатели координат точки  $P_1 + P_2$  взаимно просты с  $n$  тогда и только тогда, когда  $\forall p|n \quad P_1 \pmod{p} + P_2 \pmod{p} \neq P_\infty \pmod{p}$ .

**Доказательство.** Необходимость. Пусть у всех трех точек  $P_1 = (x_1, y_1), P_2 = (x_2, y_2), P_1 + P_2 = (x_3, y_3)$  знаменатели координат взаимно просты с  $n$ . Пусть  $p|n$ .

1. Пусть  $x_1 \neq x_2 \pmod{p}$ , тогда  $P_1 \pmod{p} + P_2 \pmod{p} \neq P_\infty \pmod{p}$  (из формул сложения).

2. Пусть  $x_1 \equiv x_2 \pmod{p}$ .

2.1. Пусть  $P_1 = P_2$ . Тогда знаменатель в  $x_3$  и  $y_3$  равен  $2y_1$ . Если  $2y_1$  делится на  $p$ , то и  $3x_1^2 + a$  делится на  $p$ , то есть  $x_1$  – корень и полинома  $x^3 + Ax + B$ , и его производной  $3x^2 + A$  – противоречие.

2.2. Пусть  $P_1 \neq P_2$ . Из  $x_1 \equiv x_2 \pmod{p}$ ,  $x_1 \neq x_2$  следует, что  $x_2 = x_1 + p^t x$ , где  $t \geq 1$  таково, что ни числитель, ни знаменатель числа  $x$  не делятся на  $p$ . Тогда  $y_2 = y_1 + p^t y$  (из формул сложения) и при этом

$$y_2^2 = (x_1 + p^t x)^3 + A(x_1 + p^t x) + B \equiv x_1^3 + Ax_1 + B + p^t x(3x_1^2 + A) = y_1^2 + p^t x(3x_1^2 + A) \pmod{p^{t+1}}. \quad (*)$$

Из  $x_1 \equiv x_2 \pmod{p}$ ,  $y_1 \equiv y_2 \pmod{p}$  имеем  $P_1 \pmod{p} = P_2 \pmod{p}$ , то есть  $P_1 \pmod{p} + P_2 \pmod{p} = 2P_1 \pmod{p}$ . Равенство  $2P_1 \pmod{p} = P_\infty \pmod{p}$  равносильно сравнению  $y_1 \equiv 0 \pmod{p}$ , то есть  $y_2^2 - y_1^2 = (y_2 - y_1)(y_2 + y_1)$  делится на  $p^{t+1}$ . Тогда из  $(*)$  следует, что  $3x_1^2 + A \equiv 0 \pmod{p}$  – противоречие, как в п. 2.1, то есть  $P_1 \pmod{p} + P_2 \pmod{p} \neq P_\infty \pmod{p}$ .

Достаточность. Пусть  $\forall p|n$  выполняется  $P_1(\text{mod } p) + P_2(\text{mod } p) \neq P_\infty(\text{mod } p)$ . Покажем, что знаменатели в  $x_3$  и  $y_3$  взаимно просты с  $n$ , то есть что эти знаменатели не делятся ни на какой  $p|n$ . Зафиксируем какое-нибудь  $p$ .

3. Пусть  $x_1 \neq x_2 \pmod{p}$ , тогда знаменатели не делятся на  $p$  (из формул сложения).
4. Пусть  $x_1 \equiv x_2 \pmod{p}$ . Тогда  $y_1 \equiv y_2 \pmod{p}$ , причем  $y_1 \equiv y_2 \neq 0 \pmod{p}$ . (Ситуацию  $y_1 \equiv -y_2 \pmod{p}$  отбрасываем, так как  $P_1(\text{mod } p) + P_2(\text{mod } p) \neq P_\infty(\text{mod } p)$ ).
- 4.1. При  $P_1 = P_2$  из формулы удвоения получаем, что  $2y_1 \neq 0 \pmod{p}$ , то есть  $\text{НОД}(2y_1, n) = 1$ .
- 4.2. При  $P_1 \neq P_2$ , как и в п. 2.2, получаем  $x_2 = x_1 + p^t x$  и из (\*):  $\frac{y_2^2 - y_1^2}{x_2 - x_1} \equiv 3x_1^2 + A \pmod{p}$ . Кроме того, число  $y_1 + y_2 \equiv 2y_1 \pmod{p}$  не делится на  $p$ , тогда и знаменатель числа  $\frac{y_2^2 - y_1^2}{(y_1 + y_2)(x_2 - x_1)} = \frac{y_2 - y_1}{x_2 - x_1}$  (и знаменатель координат точки  $P_1 + P_2$ ) не делится на  $p$ . □

Пусть  $P$  – точка кривой  $E(\mathbf{Z}/n\mathbf{Z})$ . Пусть  $D$  – база разложения, содержащая простые числа  $p_i$ , меньшие заданной границы  $m$ . Найдем целочисленное произведение  $k = \prod_{p_i \in D} p_i^{\alpha_i}$  всех элементов базы  $D$ , где

$$\alpha_i = \left\lfloor 0,5 \frac{\log n}{\log p_i} \right\rfloor.$$

### Алгоритм. Разложение числа на эллиптической кривой

*Вход.* Число  $n$ , размер  $m$  базы  $D$ .

*Выход.* Нетривиальный делитель  $d$  числа  $n$ .

1. Выбрать случайную кривую  $E(\mathbf{Z}/n\mathbf{Z})$  и точку  $P$  на ней.
2. Положить  $i \leftarrow 0, P_i \leftarrow P$ .
3. При  $i > m$  вернуться на шаг 1. В противном случае найти  $i$ -е простое число  $p_i$ .
4. Положить  $i \leftarrow i + 1, \alpha_i = \left\lfloor 0,5 \frac{\log n}{\log p_i} \right\rfloor, j \leftarrow 0$ .
5. При  $j > \alpha_i$  перейти на шаг 3. В противном случае выполнить следующие действия.
  - 5.1. Положить  $P_i \leftarrow p_i P_i$ . При каждом сложении точек вычислять  $d = \text{НОД}(n, \lambda)$ , где  $\lambda$  – угловой коэффициент касательной (секущей). При  $1 < d < n$  результат:  $d$ .
  - 5.2. Положить  $j \leftarrow j + 1$  и вернуться на шаг 5.
6. Если нетривиальный делитель числа  $n$  не найден, то вернуться на шаг 1. ■

Достаточно не формировать базу  $D$ , а находить очередной ее элемент и умножать на него текущую точку. Поэтому здесь возможно неограниченное распараллеливание с помощью невзаимодействующих компьютеров, каждый из которых работает со своим семейством эллиптических кривых.

## 21. Доказательства с нулевым разглашением.

**Доказывающий** (prover, P) — утверждает, что знает решение некоторой задачи

**Проверяющий** (verifier, V) — знает условие задачи и выполняет проверку знания

**Доказательство с нулевым разглашением** (zero-knowledge proof): информация, получаемая V от P, может быть создана им самим с полиномиальной сложностью без взаимодействия с P, то есть V не получает (а P не разглашает) никакой содержательной информации (знаний) о решении указанной задачи.

«Доказательство» носит вероятностный характер и означает, что утверждение имеет место с некоторой вероятностью, которая может быть выбрана сколь угодно близкой к единице.

### Доказательство знания разложения составного числа

Задача разложения целого числа на множители:  $n = pq$ .

Проверяющий V знает число  $n$ . Доказывающий P знает  $p$  и  $q$ .

**Протокол 1.** Доказательство знания разложения числа  $n = pq$ .

Вход V. Составное число  $n$ ; число повторов  $k$ .

Вход P. Делители  $p, q$  числа  $n$ ; число повторов  $k$ .

Выход V. Решение о том, что P знает разложение числа  $n$ , с вероятностью ошибки не более  $2^{-k}$ .

1. V и P полагают  $i \leftarrow 1$ .
2. V генерирует число  $x$ , вычисляет  $y \leftarrow x^2 \pmod{n}$  и направляет P значение  $y$ .
3. P генерирует случайный показатель  $m$ , вычисляет  $z \leftarrow y^m \pmod{n}$ , вычисляет квадратный корень из  $\sqrt{z} \pmod{n}$  и направляет V число  $z$ .
4. V генерирует случайный бит и направляет его P.

2

- 
5. Если P получает 0, то он раскрывает показатель  $m$ ; если получает 1, то раскрывает вычисленное значение квадратного корня.

6. V проверяет правильность полученного ответа.

Если ответ не совпадает, то результат: P не знает разложения.

Иначе:

Если  $i < k$ , то V сообщает P о правильности этого шага; V и P полагают  $i \leftarrow i + 1$  и возвращаются к шагу 2.

Иначе: результат: P знает разложение.

## Доказательство знания дискретного логарифма

Задача дискретного логарифмирования в группе  $G = \langle a \rangle$ :  $b = a^x$ .

Проверяющий V знает  $b$ . Доказывающий P знает  $x$ .

**Протокол 2.** Доказательство знания дискретного логарифма в циклической группе  $G$ .

Вход V. Группа  $G$ , образующая  $a$  и ее образ  $b$ ; число повторов  $k$ .

Вход P. Логарифм  $x$  такой, что  $b = a^x$ ; число повторов  $k$ .

Выход V. Решение о том, что P знает логарифм, с вероятностью ошибки не более  $2^{-k}$ .

1. V и P полагают  $i \leftarrow 1$ .

2. P генерирует случайное число  $y$ , вычисляет  $c \leftarrow a^y$  и направляет V значение  $c$ .

3. V проверяет, что  $c \in G$ , генерирует случайный бит и направляет его P.

4. Если P получает 0, то он раскрывает логарифм  $\log_a c \equiv y \pmod{\#G}$ ; если получает 1, то

раскрывает логарифм  $\log_b c \equiv yx^{-1} \pmod{\#G}$ .

4

5. V проверяет правильность полученного логарифма и его соответствие биту, выработанному на шаге 3 (должно выполняться равенство  $c = a^y$  для «0» и  $c = b^{yx^{-1}}$  для «1»).

Если соответствующее равенство не выполняется, то результат: P не знает логарифма.

Иначе:

Если  $i < k$ , то V сообщает P о правильности этого шага; V и P полагают  $i \leftarrow i + 1$  и возвращаются к шагу 2

Иначе результат: P знает логарифм. ■

## Протокол 3. Бездиалоговое доказательство знания логарифма.

Вход V. Группа  $G = \langle a \rangle$ ,  $b \in G$ , число  $k$ , хэш-функция  $h$ .

Вход P. Группа  $G = \langle a \rangle$ ,  $r = \#G$ , логарифм  $x$ , число  $k$ , хэш-функция  $h$ .

Выход V. Решение о том, что P знает логарифм, с вероятностью не более  $2^{-k}$ .

1. P вырабатывает  $k$  случайных показателей  $y_1, \dots, y_k$ , обратимых по модулю  $r$ , и вычисляет  $a^{y_1}, \dots, a^{y_k}$ .

2. P вычисляет хэш-функцию от аргумента, представляющего собой конкатенацию найденных экспонент:  $e \leftarrow h(a^{y_1} \| \dots \| a^{y_k})$ .

3. P посыпает V текст доказательства — набор  $(a^{y_1}, \dots, a^{y_k}; z_1, \dots, z_k)$ ,

где  $z_i = y_i$ , если  $e_i = 0$ ;  $z_i = y_i x^{-1}$ , если  $e_i = 1$ ;  $(e_1, \dots, e_k)$  — младшие  $k$  битов числа  $e$ .

4. V вычисляет хэш-функцию  $e \leftarrow h(a^{y_1} \| \dots \| a^{y_k})$ , находит биты  $e_i$  и проверяет их соответствие предъявленному тексту. При  $e_i = 0$  должно выполняться равенство  $a^{z_i} = a^{y_i}$ ; при  $e_i = 1$  — равенство  $b^{z_i} = a^{y_i}$ .

5. Если все  $k$  проверок на шаге 4 проходят успешно, то результат: P знает логарифм, иначе результат: P не знает логарифм. ■

## 22. Ранцевые алгоритмы шифрования с ОК

**Задача об укладке ранца:** дано множество натуральных чисел  $W = \{w_1, \dots, w_n\}$ . Можно ли данное натуральное число  $N$  представить в виде суммы слагаемых из некоторого подмножества множества  $W$ ?

Иначе: существует ли двоичный вектор  $(x_1, \dots, x_n)$ , такой, что  $N = \sum_{i=1}^n x_i w_i$ ? Эту задачу иногда называют **задачей о сумме подмножеств** (subset sum problem).

**Плотность ранца** — вещественное число

$$\frac{n}{\max\{\log_2 w_i | 1 \leq i \leq n\}}.$$

Непосредственно из определения следует, что плотность ранца всегда меньше 1.

1

---

Последовательность  $\{b_1, \dots, b_n\}$ , где  $b_i \in \mathbf{Z}$ , — **сверхвозрастающая**, если  $b_k > \sum_{i=1}^{k-1} b_i$  для всех  $k \leq n$ .

Решение задачи об укладке ранца для сверхвозрастающей последовательности:

Сравнить число  $N$  с  $b_n$ .

Если  $N \geq b_n$ , то положить  $x_n = 1$ ,

иначе  $x_n = 0$ .

Найти  $N - b_n x_n$ , определить  $x_{n-1}$  и т. д.

Первая крипtosистема:

Merkle R., Hellman M. Hiding information and signatures in trapdoor knapsacks // IEEE Transactions on Information Theory. 1978. Vol. IT-24. P. 525–530.

## Алгоритм 1. Генерация ключа для ранцевого шифра Меркла—Хеллмана.

*Вход.* Длина блока  $n$ .

*Выход.* Открытый и секретный ключи.

1. Выбрать сверхвозрастающую последовательность из  $n$  натуральных чисел:  $\{b_1, \dots, b_n\}$ .
2. Выбрать целое число  $u > \{b_1 + \dots + b_n\}$ .
3. Выбрать случайное число  $v$ ,  $\text{НОД}(u, v) = 1$ .
4. Выбрать случайную перестановку  $\pi \in S_n$ .
5. Вычислить последовательность  $\{a_1, \dots, a_n\}$ , где  $a_i \leftarrow vb_{\pi(i)} \pmod{u}$ .
6. Результат:  $\{a_1, \dots, a_n\}$  — открытый ключ;  $(u, v, \pi, \{b_1, \dots, b_n\})$  — секретный ключ. ■

## Протокол 1. Ранцевый шифр Меркла—Хеллмана.

*Вход отправителя.* Открытый ключ  $\{a_1, \dots, a_n\}$  из алгоритма 1.

*Вход получателя.* Секретный ключ  $(u, v, \pi, \{b_1, \dots, b_n\})$  из алгоритма 1.

Для зашифрования сообщения длины  $n$  бит,  $\mathbf{m} = (m_1, \dots, m_n)$ , отправитель вычисляет сумму

$$c \leftarrow \sum_{i=1}^n m_i a_i.$$

Для расшифрования шифртекста  $c$  получатель

1. Вычисляет  $d \leftarrow cv^{-1} \pmod{u}$ .
2. Находит представление  $d = \sum_{i=1}^n r_i b_i$ .
3. Вычисляет вектор  $\mathbf{m} = (m_1, \dots, m_n)$ , где  $m_i \leftarrow \pi(r_i)$ . ■

Корректность протокола:

$$d \equiv cv^{-1} \equiv \sum_{i=1}^n m_i a_i v^{-1} \equiv \sum_{i=1}^n m_i b_{\pi(i)} \pmod{u}.$$

**Решетка** в  $\mathbb{R}^n$  — множество  $\Lambda = \Lambda(\mathbf{b}_1, \dots, \mathbf{b}_n) = \{z_1 \mathbf{b}_1 + \dots + z_n \mathbf{b}_n \mid z_i \in \mathbb{Z}, i = 1, 2, \dots, n\}$ , где линейно независимые векторы  $\mathbf{b}_1, \dots, \mathbf{b}_n \in \mathbb{R}^n$  — **базис решетки**.

**Процесс ортогонализации** Грама—Шмидта: векторы

$$\mathbf{b}'_1 = \mathbf{b}_1, \quad \mathbf{b}'_i = \mathbf{b}_i - \sum_{j=1}^{i-1} \mu_{i,j} \mathbf{b}'_j, \quad i = 2, \dots, n,$$

где  $\mu_{i,j} = \frac{(\mathbf{b}_i, \mathbf{b}'_j)}{(\mathbf{b}'_j, \mathbf{b}'_j)}$ ,  $1 \leq j < i$ , попарно ортогональны и порождают то же линейное пространство, что и  $\mathbf{b}_1, \dots, \mathbf{b}_n$ .

Базис  $\{\mathbf{b}_1, \dots, \mathbf{b}_n\}$  — **LLL-приведенный** (Ленстра, Ленстра, Ловас), если

$$|\mu_{i,j}| \leq \frac{1}{2}, \quad 1 \leq j < i \leq n,$$

$$\|\mathbf{b}'_i + \mu_{i,j} \mathbf{b}'_{i-1}\|^2 \geq \frac{3}{4} \|\mathbf{b}'_{i-1}\|^2.$$

Основное свойство LLL-приведенного базиса: для любой и ненулевой точки решетки  $\mathbf{x} \in \Lambda \setminus \{\mathbf{0}\}$  справедливо неравенство  $\|\mathbf{b}_1\|^2 \leq 2^{n-1} \|\mathbf{x}\|^2$ .

Дешифрование крипtosистемы Меркля—Хеллмана.

Вход: открытый ключ  $\{a_1, \dots, a_n\}$ , шифртекст  $c$ .

Выход: вектор  $(x_1, \dots, x_n) \in \{0,1\}^n$ , такой что  $c = \sum_{i=1}^n x_i a_i$ , либо «решение не найдено».

1. Положить  $t \leftarrow \left\lfloor \frac{1}{2} \sqrt{n} \right\rfloor$ .

2. Вычислить LLL-приведенный базис решетки  $\Lambda$ , заданной строками матрицы

$$\begin{pmatrix} 1 & 0 & \dots & 0 & ta_1 \\ 0 & 1 & \dots & 0 & ta_2 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & ta_n \\ \frac{1}{2} & \frac{1}{2} & \dots & \frac{1}{2} & tc \end{pmatrix}.$$

3. Для каждого вектора  $\mathbf{b} = (b_1, \dots, b_{n+1})$  приведенного базиса с  $b_{n+1} = 0$  и  $|b_i| = \frac{1}{2}$  для всех  $i = 1, \dots, n$ :

3.1 Для всех  $i = 1, 2, \dots, n$  положить  $x_i \leftarrow b_i + \frac{1}{2}$ . Если  $\sum_{i=1}^n x_i a_i = c$ , то результат:  $(x_1, \dots, x_n)$ .

3.2 Для всех  $i = 1, 2, \dots, n$  положить  $x_i \leftarrow -b_i + \frac{1}{2}$ . Если  $\sum_{i=1}^n x_i a_i = c$ , то результат:  $(x_1, \dots, x_n)$ .

4. Результат «решение не найдено».

Если  $\mathbf{c}_i$  — элемент базиса решетки  $\Lambda$  из алгоритма дешифрования и  $(x_1, \dots, x_n) \in \{0,1\}^n$  такой, что  $\sum_{i=1}^n x_i a_i = c$ , то  $\mathbf{b} = (b_1, \dots, b_{n+1}) = \sum_{i=1}^n x_i \mathbf{c}_i - \mathbf{c}_{n+1}$  принадлежит  $\Lambda$ , при этом  $b_{n+1} = 0$ ,  $|b_i| = \frac{1}{2}$  для всех  $i = 1, 2, \dots, n$  и  $\|\mathbf{b}\| = \sqrt{b_1^2 + \dots + b_{n+1}^2} = \sqrt{\frac{n}{4} + 0} = \frac{\sqrt{n}}{2}$ , то есть  $\mathbf{b}$  — самый короткий ненулевой вектор решетки), и если плотность ранца не превосходит 0,9408, то алгоритм решает задачу об укладке ранца с вероятностью, близкой к 1.

## Алгоритм 2. Генерация ключа для ранцевого шифра Шора—Ривеста.

Вход. Характеристика поля  $p$ .

Выход. Открытый и секретный ключи.

1. Выбрать конечное поле  $\mathbf{F}_q = \mathbf{F}_p[x]/(f(x))$  из  $q = p^h$  элементов, задаваемое неприводимым над  $\mathbf{F}_p$  полиномом  $f(x)$  и такое, что задача логарифмирования в  $\mathbf{F}_q$  не является сложной.

2. Выбрать образующую  $g(x)$  группы  $\mathbf{F}_q^*$ .

3. Для  $i = 0, 1, \dots, p-1$  вычислить логарифмы  $a_i \leftarrow \log_{g(x)}(x+i)$ .

4. Выбрать случайную перестановку  $\pi \in S_p$ .

5. Выбрать случайное число  $d, 0 \leq d \leq q-2$ .

6. Для  $i = 0, 1, \dots, p-1$  вычислить  $c_i \leftarrow (a_{\pi(i)} + d) \pmod{(q-1)}$ .

7. Результат:  $(\{c_0, \dots, c_{p-1}\}, p, h)$  — открытый ключ;  $(f(x), g(x), \pi, d)$  — секретный ключ. ■

**Протокол 2.** Ранцевый шифр Шора—Ривеста.

*Вход отправителя.* Открытый ключ  $(\{c_0, \dots, c_{p-1}\}, p, h)$  из алгоритма 2.

*Вход получателя.* Секретный ключ  $(f(x), g(x), \pi, d)$  из алгоритма 2.

Для зашифрования сообщения  $m$ ,  $0 \leq m < C_p^h$ , отправитель:

1. Преобразует сообщение  $m$  в двоичный вектор  $\mathbf{m} = (m_0, \dots, m_{p-1})$ , содержащий ровно  $h$  единиц:

1.1.  $l \leftarrow h$ .

1.2. Для  $i = 1, 2, \dots, p$  полагает  $m_{i-1} \leftarrow 1, m \leftarrow m - C_{p-i}^l, l \leftarrow l - 1$  при  $m \geq C_{p-i}^l$ ; иначе

$$m_{i-1} \leftarrow 0.$$

2. Вычисляет шифртекст

$$c \leftarrow \sum_{i=0}^{p-1} m_i c_i \pmod{(q-1)}.$$

8

Для расшифрования шифртекста  $c$  получатель:

1. Вычисляет  $r \leftarrow c - hd \pmod{(q-1)}$ .
2. Находит  $u(x) \leftarrow g(x)^r \pmod{f(x)}$ .
3. Вычисляет полином  $s(x) \leftarrow u(x) + f(x)$  над  $\mathbb{F}_p$  и раскладывает его на линейные множители:  $s(x) = \prod_{j=1}^h (x + t_j)$ .
4. Находит единичные компоненты вектора  $\mathbf{m}$  (стоят на позициях с номерами  $\pi^{-1}(t_j)$ ).

Остальные компоненты вектора  $\mathbf{m}$  — нулевые.

5. Восстанавливает сообщение  $m$ :

5.1. Полагает  $m \leftarrow 0, l \leftarrow h$ .

5.2. Для  $i = 1, 2, \dots, p$  полагает  $m \leftarrow m + C_{p-i}^l, l \leftarrow l - 1$  при  $m_{i-1} = 1$ . ■

Корректность протокола:

$$\begin{aligned} u(x) &\equiv g(x)^r = g(x)^{c-hd} = g(x)^{\left(\sum_{i=0}^{p-1} m_i c_i\right) - hd} = g(x)^{\left(\sum_{i=0}^{p-1} m_i(a_{\pi(i)}+d)\right) - hd} = \\ &= g(x)^{\sum_{i=0}^{p-1} m_i a_{\pi(i)}} = \prod_{i=1}^p (g(x)^{a_{\pi(i)}})^{m_i} \equiv \prod_{i=1}^p (x + \pi(i))^{m_i} \equiv s(x) \pmod{f(x)}. \end{aligned}$$