Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина)

Лабораторная работа № 8

Изучение и исследование алгоритмов создания и проверки электронной подписи

Студентка: Усачева Дарья, группа 1384

Руководитель: Племянников А.К., доцент каф. ИБ

Санкт-Петербург, 2024

Цель работы и задачи

Цель: Повысить свою компетенцию в области алгоритмов создания и проверки электронной подписи и в криптографии в целом.

Задачи:

1. Изучить генерацию ключевых пар для алгоритмов RSA, DSA, и ECDSA.
2. Изучить процессы создания и проверки электронных подписей.
3. Создать и проверить электронную подпись на основе эллиптических кривых.
4. Продемонстрировать процесс подписи в среде PKI.
5. Подписать свой отчет.

Изучение генерации ключевых пар для алгоритмов RSA, DSA, и ECDSA

Описание алгоритмов генерации ключевых пар

Генерация ключевых пар для алгоритма RSA (с использованием двух больших простых чисел p и q, которые хранятся в секрете) включает следующие шаги:

1. Вычисление n = p × q.

2. Выбор произвольного числа e (e < n), которое является взаимно простым с φ(n).

3. Определение d, удовлетворяющего уравнению e × d = 1 mod φ(n).

4. Пара (e, n) формирует открытый ключ, а d – закрытый ключ; значения p и q подлежат уничтожению.

Генерация ключевых пар для алгоритма DSA включает следующие этапы:

1. Выбор числа p с длиной 512 - 1024 бит, при этом длина p должна быть кратна 64.

2. Выбор числа q, которое имеет такой же размер в битах, как и размер дайджеста используемой хеш-функции (160 бит для SHA-1), и удовлетворяет условию

p - 1 = 0 mod q.

3. Выбор e1, для которого выполняется

$e_1^q = 1 \, mod \, p$.

4. Выбор целого числа d < q и вычисление e_2, для которого выполняется $e_2 = e_1^d mod \, p$

5. Числа (e1, e2, p, q) образует открытый ключ, а d – закрытый ключ.

Описание алгоритмов генерации ключевых пар

Генерация ключевых пар для алгоритма ECDSA включает следующие этапы:

1. Выбор эллиптической кривой E_p (a, b), где p - простое числом.

2. Выбор точки на кривой $e_1 = (x_1, y_1)$.

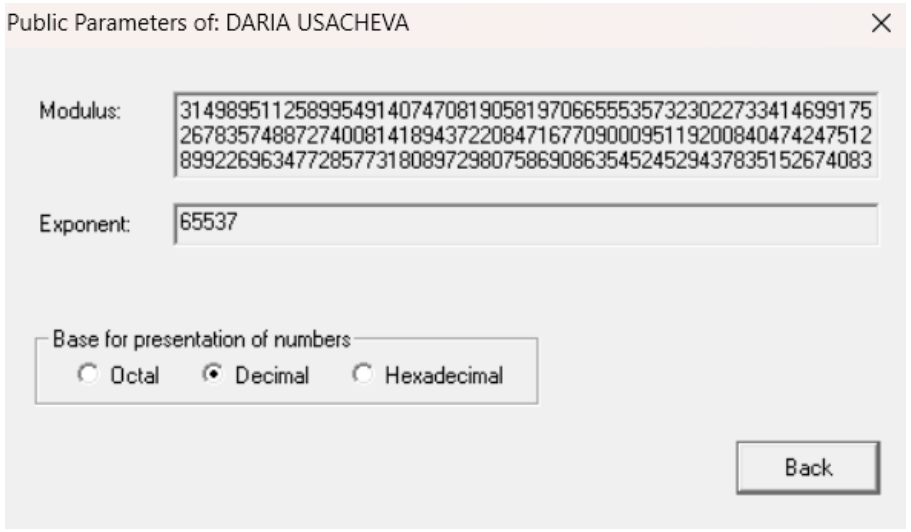
3. Выбор простого числа q - порядка одной из циклических подгрупп группы точек эллиптической кривой, удовлетворяющего условию $q \times (x_1, y_1) = 0$.

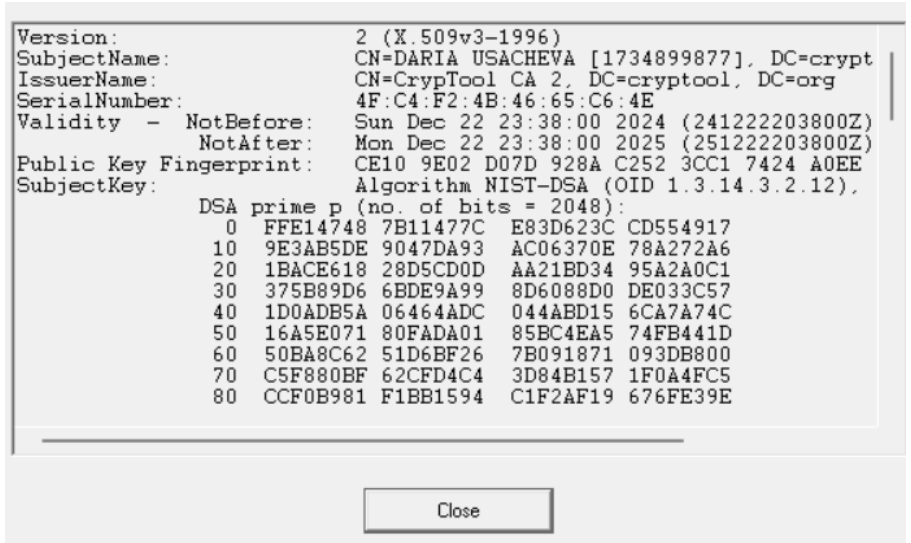
4. Выбор закрытого ключа d.

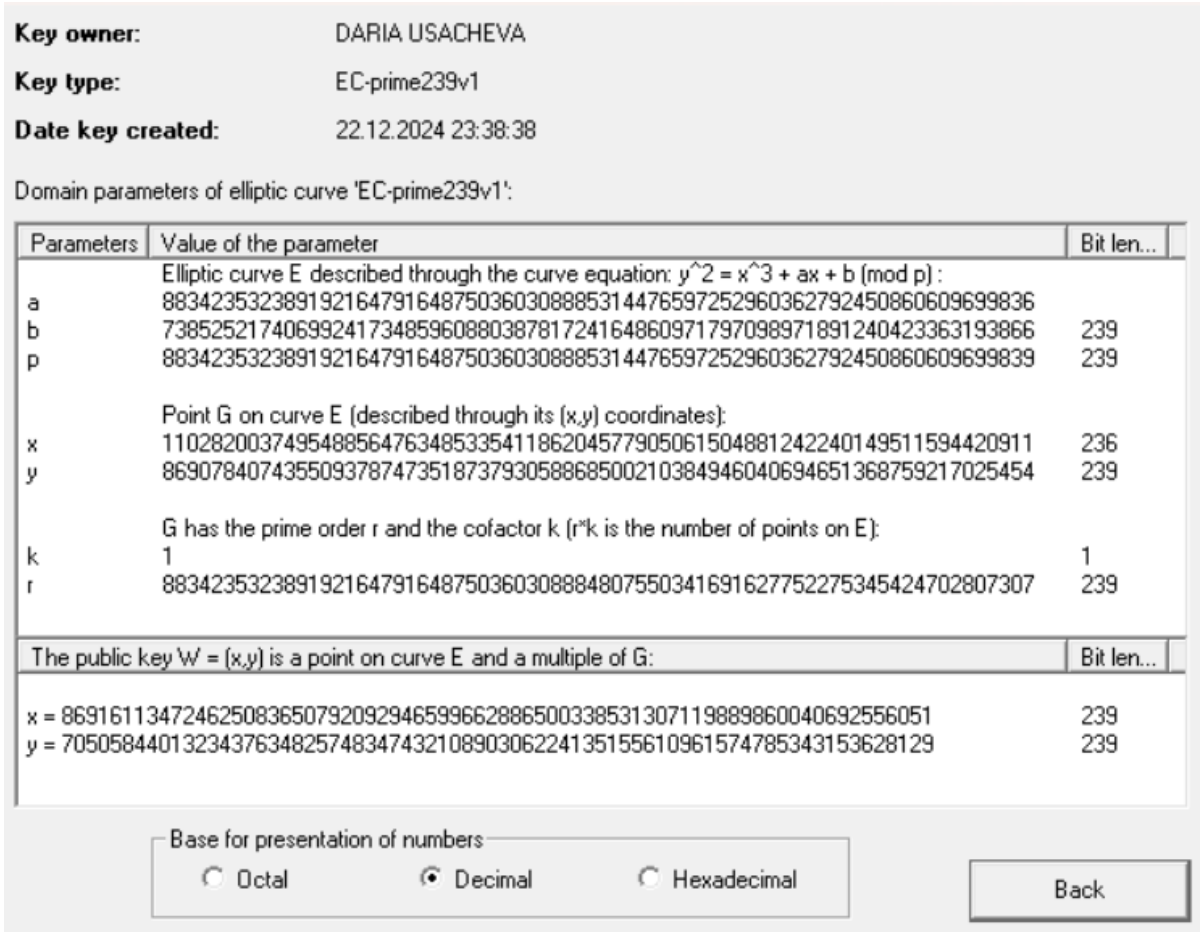
5. Вычисление точки на кривой $e_2 = d \times e_1$.

6. Открытый ключ (a, b, q, p, e_1, e_2).

Значения открытых ключей





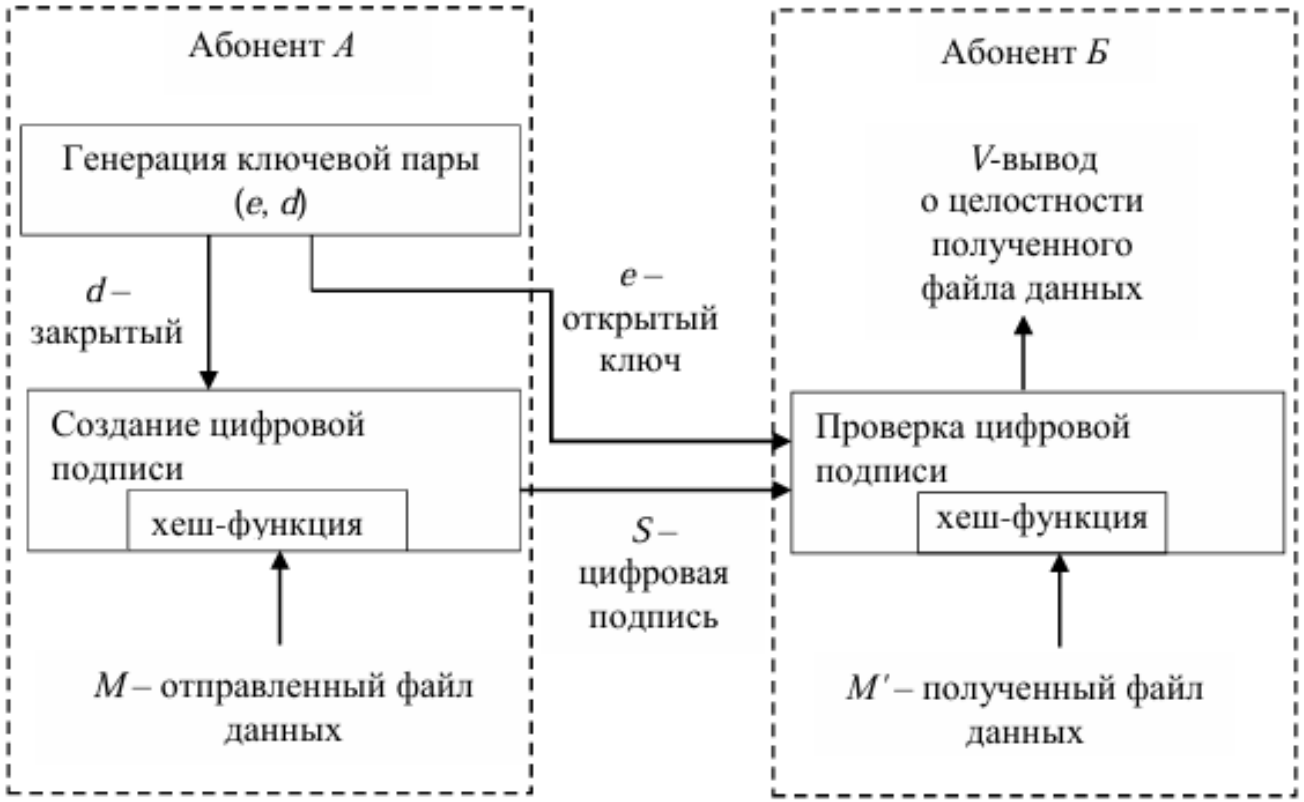


Фактическое время генерации ключевых пар

Алгоритм	Время генерации
RSA-2048	2,861 сек
DSA-2048	3,061 сек
EC-239	0,014 сек

Изучение процессов создания и проверки электронных подписей

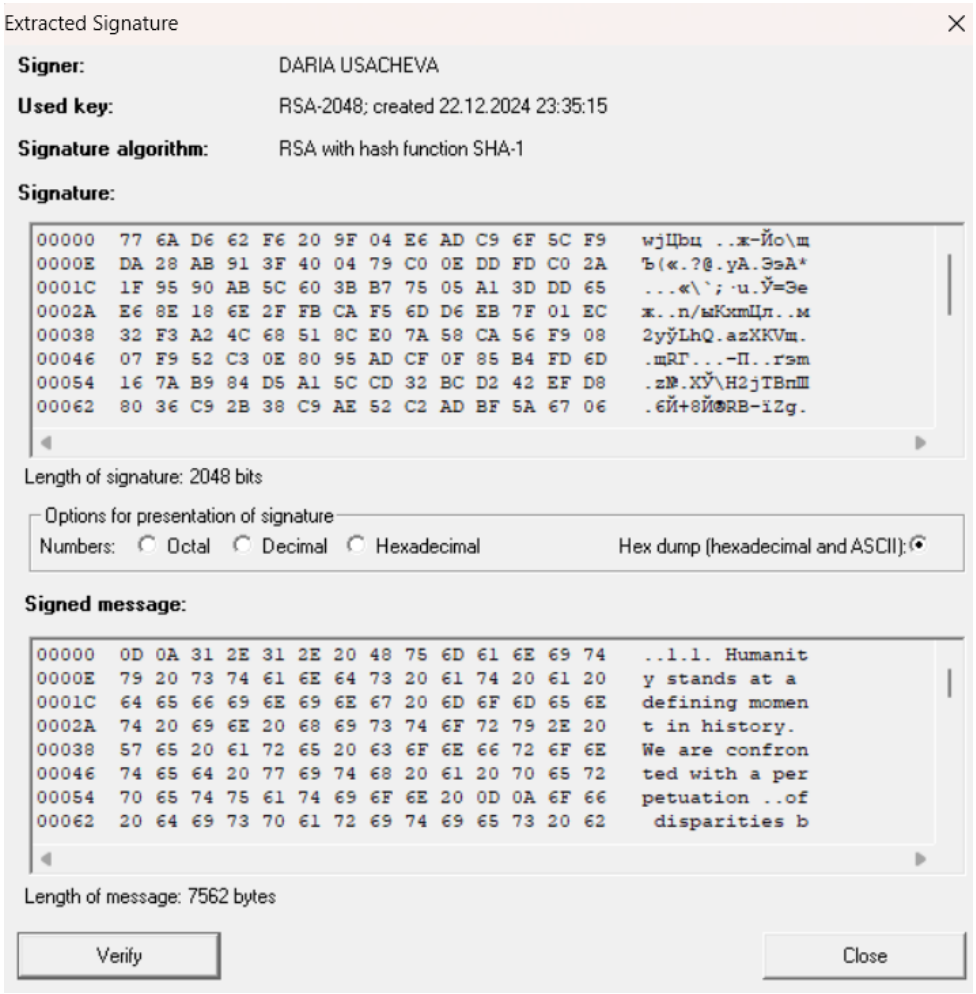
Схема, поясняющая работу протокола подписания документа и проверки электронной подписи



Фактическое время генерации электронной подписи

Алгоритм	Время генерации
RSA-2048	0,008 сек
DSA-2048	0,002 сек
EC-239	0,000 сек

Значение электронной подписи и результат ее проверки



Создание и проверка электронной подписи на основе эллиптических кривых

Алгоритмы создания и проверки электронной подписи ECDSA

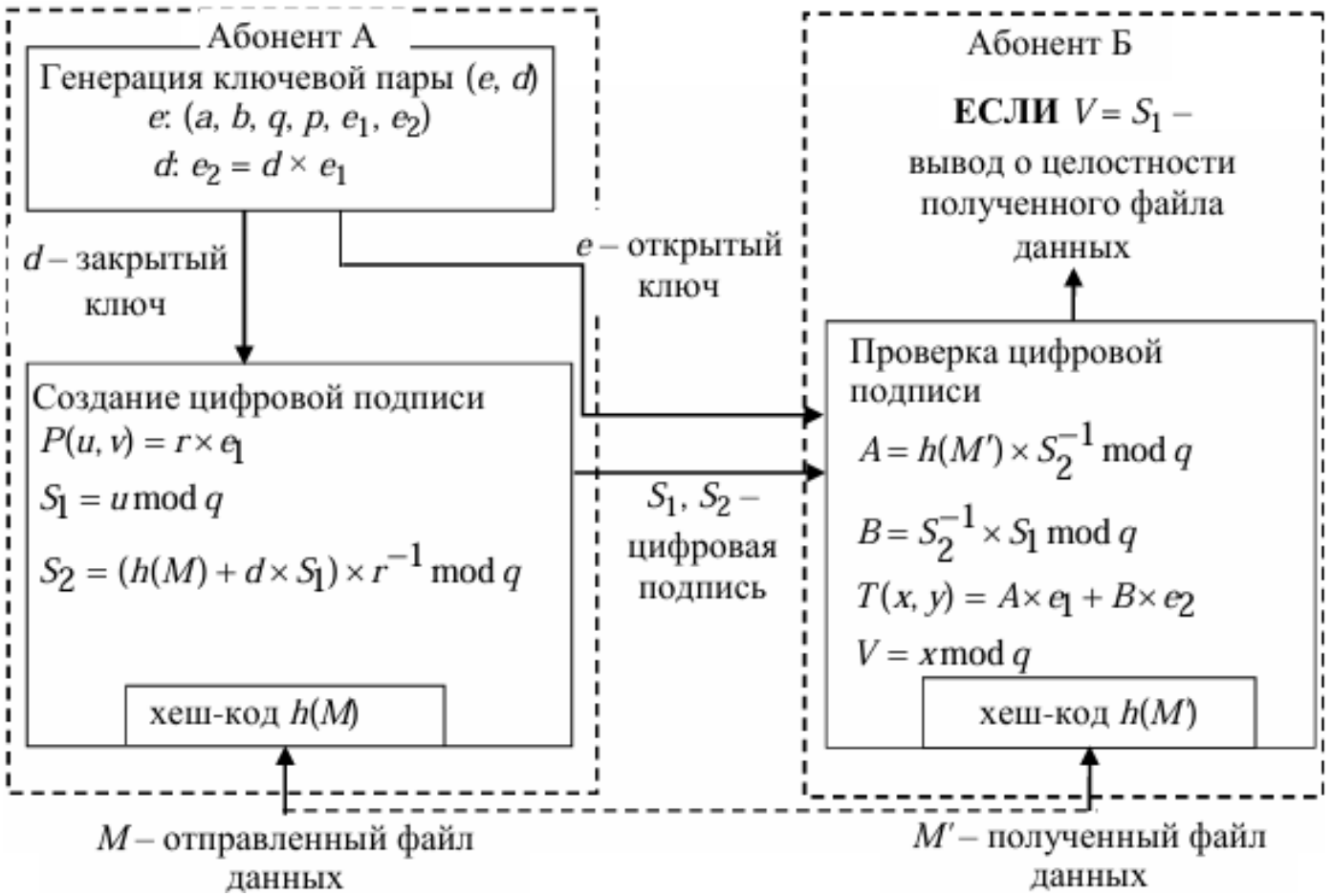
Алгоритм создания электронной подписи ECDSA

1. Выбирается секретное случайное число r: $r \in (1, q-1)$.
2. Выбирается третья точка на кривой: $P(u, v) = r \times e_1$.
3. Вычисляется первая часть подписи по формуле $S_1 = u \bmod q$, где u — абсцисса.
4. Вычисляется вторая часть подписи по формуле:
   $S_2 = (h(M) + d \times S_1) \times r^{-1} \bmod q$,
   где h(M) — хеш-код сообщения; d — закрытый ключ.

Алгоритм проверки электронной подписи ECDSA

1. Вычисляются промежуточные результаты A и B:
   $A = h(M) \times S_2^{-1} \bmod q$,
   $B = S_2^{-1} \times S_1 \bmod q$.
2. Восстанавливается третья точка:
   $T(x, y) = A \times e_1 + B \times e_2$.
   Верификатор $V = x \bmod q$ сравнивается с первой частью подписи S_1.

Подписание документа и верификация подписи



Последовательность шагов создания подписи

Message M to be signed:

```
00000  6D 65 6F 77                                        meow
```

Step-by-step signature generation:

```
Signature originator:  DARIA USACHEVA

Domain parameters to be used 'EC-prime239v1':

  a  = 8834235323891921647916487503603088853144765972529603627924508
  b  = 7385252174069924173485960880387817241648609717970989718912404
  Gx = 1102820037495488564763485335411862045779050615048812422401495
  Gy = 8690784074355093787473518737930588685002103849460406946513687
  k  = 1
  r  = 8834235323891921647916487503603088848075503416916277522753454

Secret key s of the signature originator:

  s  = 2915036748500044271323721199647829903875691411705533411701981
```

Step 0 out of a maximum of 6 steps.

Output signature data | Cancel | Continue >

Step-by-step signature generation:

```
  s  = 2915036748500044271323721199647829903875691411705533411701981

Chosen signature algorithm: ECSP-DSA with hash function SHA-1

Size of message M to be signed: 4 bytes

Continue ...

Calculate a 'hash value' f (message representative) from message M,

  f  = 715679189069181012862657191738191532500000735025

Continue ...
```

Step 1 out of a maximum of 6 steps.

Step-by-step signature generation:

```
  f  = 715679189069181012862657191738191532500000735025

Continue ...

Create a random one-time key pair (secret key, public key) = (u,V)
with the domain parameters of 'EC-prime239v1' (V=(Vx,Vy) is a point

  u  = 5071674852595975551382618374972494937870016254269350213619311
  Vx = 6730505914931207341002609671744548717026099910946475396907831
  Vy = 5567321228873950613597649967034883259668510280419585287825513

Continue ...
```

Step 2 out of a maximum of 6 steps.

Последовательность шагов создания подписи

Step-by-step signature generation:

```
with the domain parameters of 'EC-prime239v1' (V=(Vx,Vy) is a point (

  u   = 459629910345459949508463172320931166253644580288818749492551
  Vx  = 433028134336289297806474749477903849441450665220581769093522
  Vy  = 560008800154351338425267528898070975226680271039049008483920

Continue ...

Convert the group element Vx (x co-ordinates of point V on elliptic (

  i   = 433028134336289297806474749477903849441450665220581769093522

Continue ...
```

Step 3 out of a maximum of 6 steps.

Step-by-step signature generation:

```
Continue ...

Convert the group element Vx (x co-ordinates of point V on elliptic (

  i   = 433028134336289297806474749477903849441450665220581769093522

Continue ...

Calculate the number c = i mod r (c not equal to 0):

  c   = 433028134336289297806474749477903849441450665220581769093522

Continue ...
```

Step 4 out of a maximum of 6 steps.

Step-by-step signature generation:

```
Continue ...

Calculate the number c = i mod r (c not equal to 0):

  c   = 433028134336289297806474749477903849441450665220581769093522

Continue ...

Calculate the number d = u^(-1)*(f + s*c) mod r (d not equal to 0):

  d   = 538501903754592752924770011686524739407854038268298804680918

Continue ...
```

Step 5 out of a maximum of 6 steps.

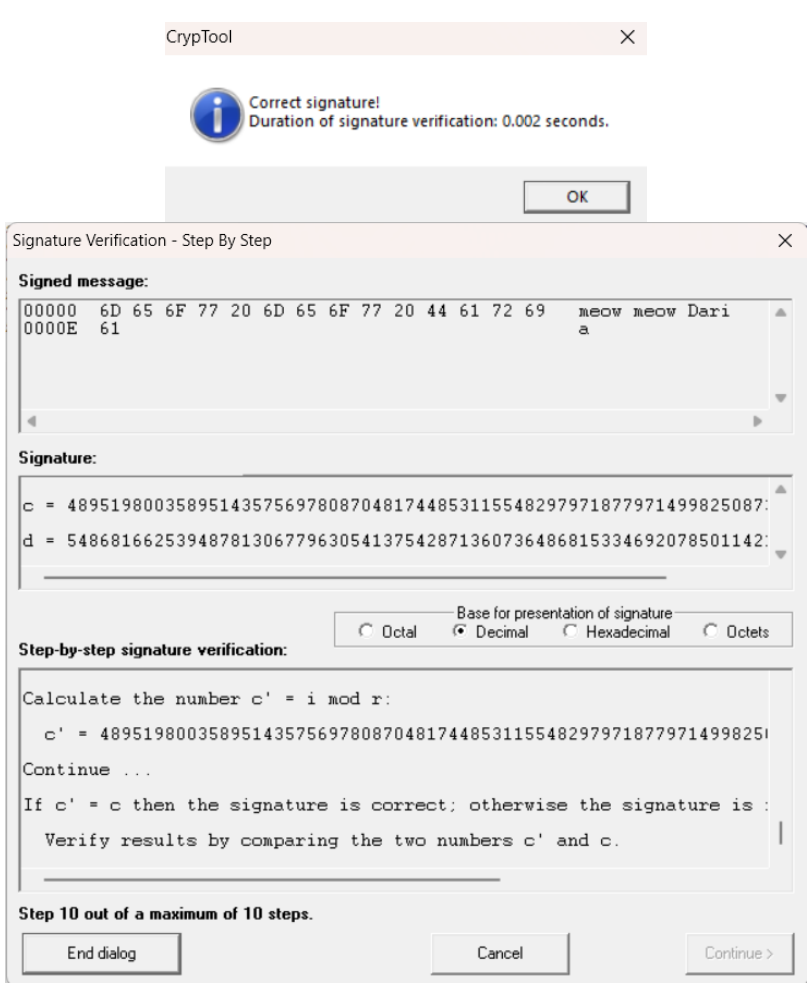
Step-by-step signature generation:

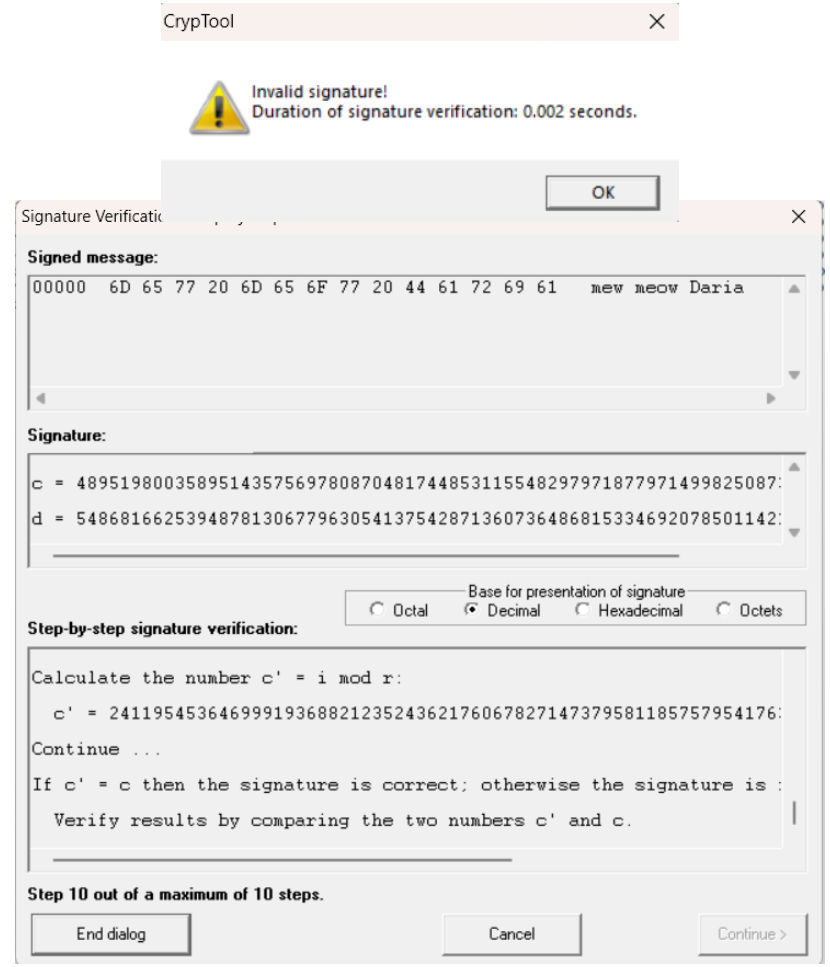
```
Calculate the number c = i mod r (c not equal to 0):

  c   = 433028134336289297806474749477903849441450665220581769093522

Continue ...

Calculate the number d = u^(-1)*(f + s*c) mod r (d not equal to 0):

  d   = 538501903754592752924770011686524739407854038268298804680918

Continue ...

Signature generation finished.
The signature consists of the two numbers c and d.
```

Step 6 out of a maximum of 6 steps.

Результат проверки электронной подписи

Результат проверки для случаев сохранения и нарушения целостности исходного текста





Проверка лекционного материала по ECDSA

Шифр Эль-Гамаля на эллиптических кривых

1. Выбираем эллиптическую кривую E_{67} (2, 3), где p - простое числом.

2. Выбираем точку на кривой e_1 = (2, 22).

3. Выбираем закрытый ключ d = 4.

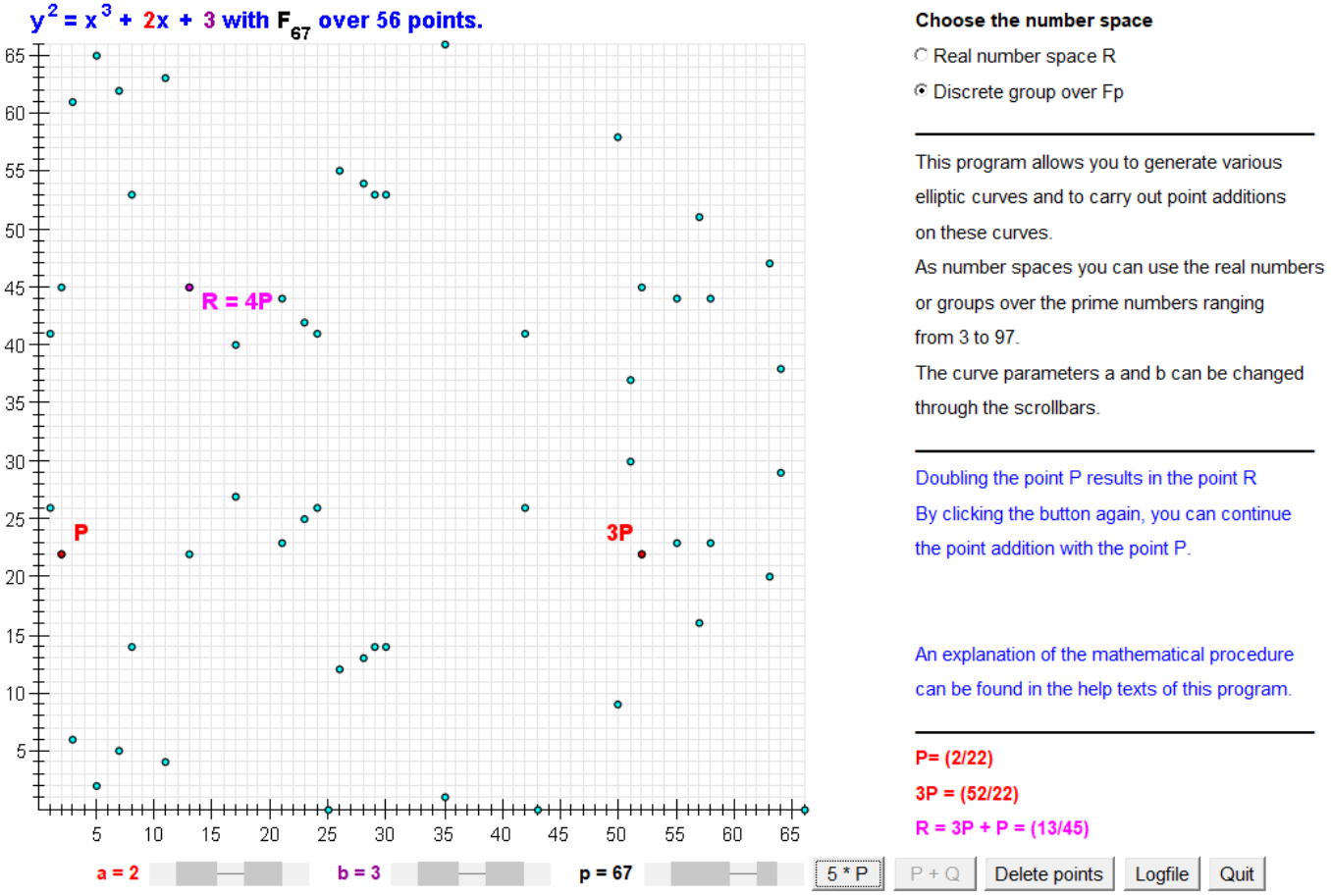
4. Вычисление точки на кривой e_2 = d × e_1 = 4 × (2, 22) = (13, 45).

6. Открытый ключ (2, 3, 67,(2, 22),(13, 45)).

7. Выбираем точку для сопоставления открытому тексту P = (24, 26) на кривой.
8. Выбираем секретное случайное число r = 2: r ∈ (1, q-1).
9. Создаем шифровку C_1, C_2:
$C_1 = r \times e_1$ = 2 × (2, 22) = (35,1)
$C_2 = P + r \times e_2$ = (24, 26) + 2 × (13,45) = (24, 26) + (23, 25) = (21, 44)
10. Получатель выполняет расшифровку:
$C_2 - (d \times C_1) = P + r \times d \times e_1 - d \times r \times e_1$ = P
(21, 44) – (4 × (35,1)) = (21, 44) + (23, 42) = (24, 26)

- (4 × (35,1)) = – (23, 25) = (23, 42)

Проверка лекционного материала по ECDSA. Генерация ключа



Choose the number space

- ○ Real number space R
- ◉ Discrete group over Fp

This program allows you to generate various elliptic curves and to carry out point additions on these curves.

As number spaces you can use the real numbers or groups over the prime numbers ranging from 3 to 97.

The curve parameters a and b can be changed through the scrollbars.

Doubling the point P results in the point R

By clicking the button again, you can continue the point addition with the point P.

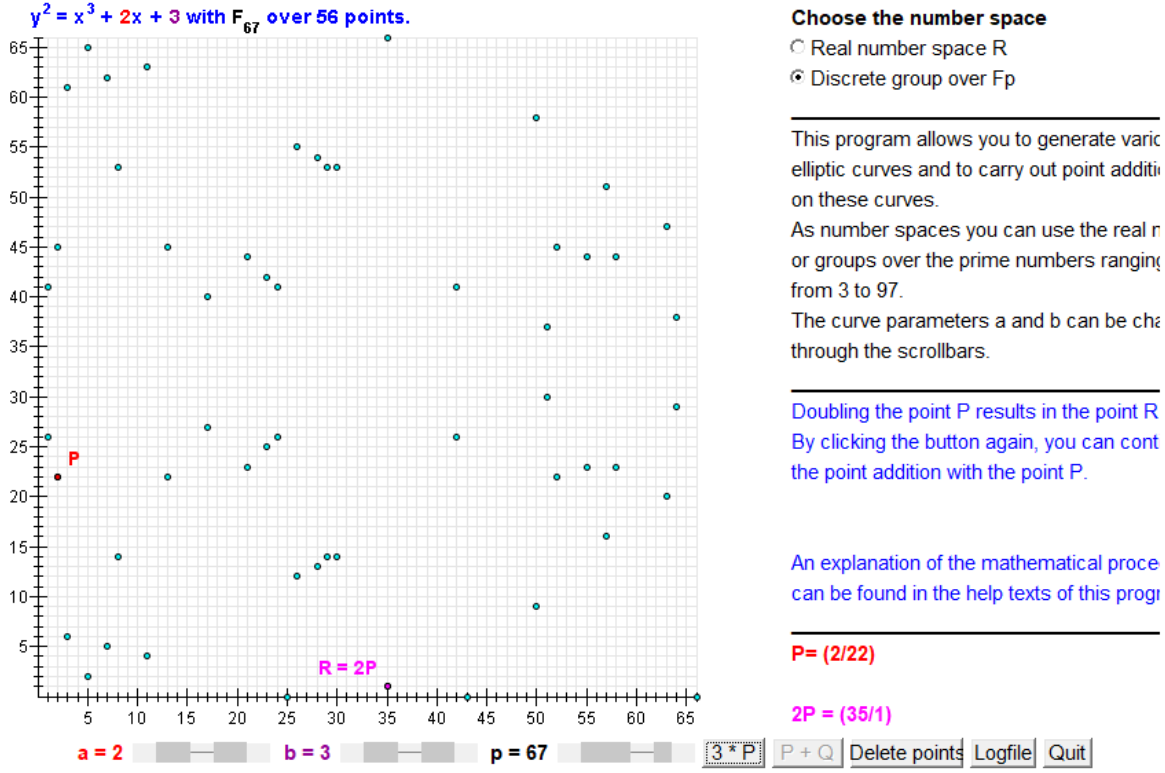
An explanation of the mathematical procedure can be found in the help texts of this program.

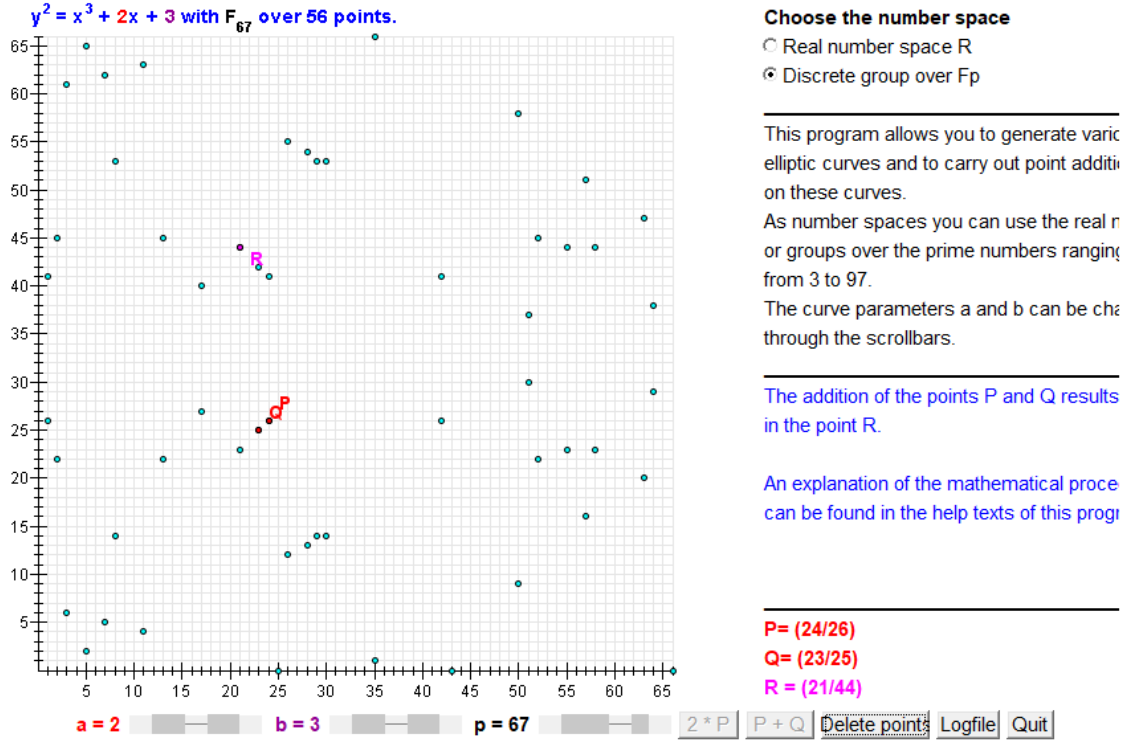
P= (2/22)

3P = (52/22)

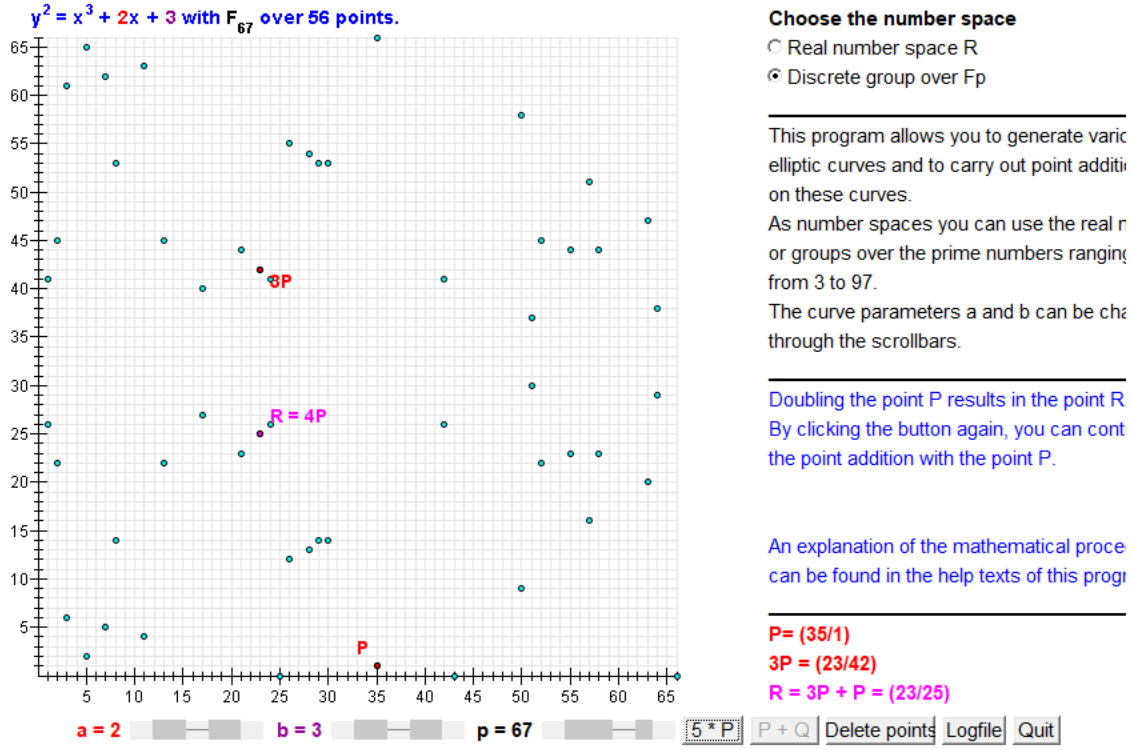
R = 3P + P = (13/45)

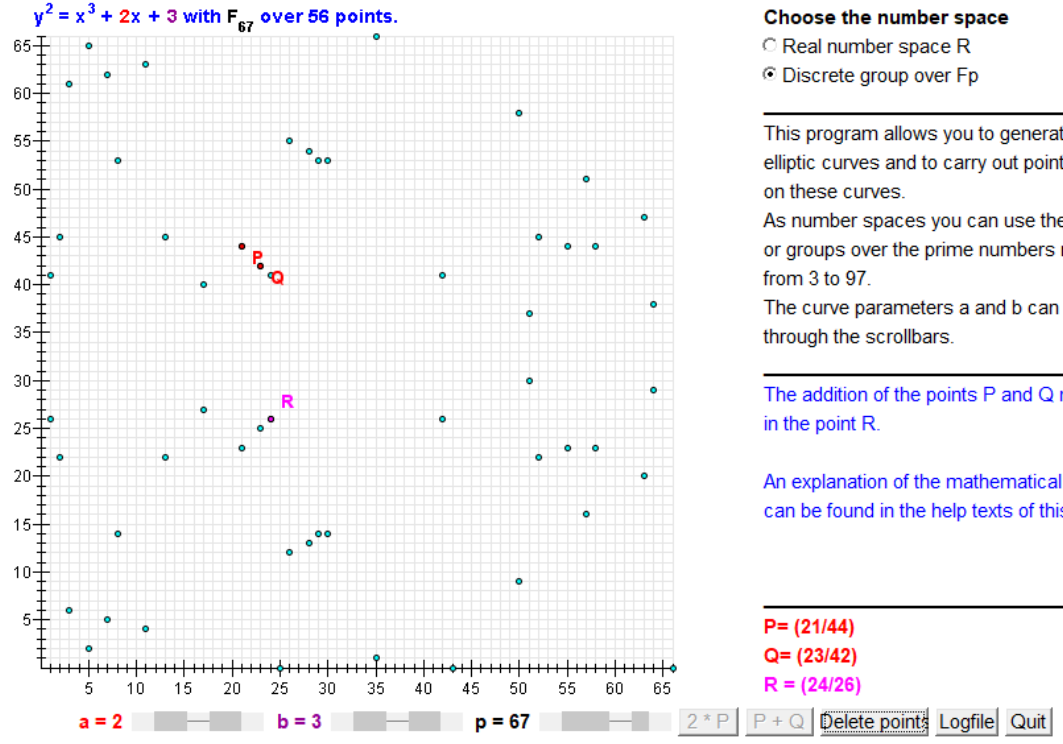
Проверка лекционного материала по ECDSA. Зашифрование





Проверка лекционного материала по ECDSA. Расшифрование

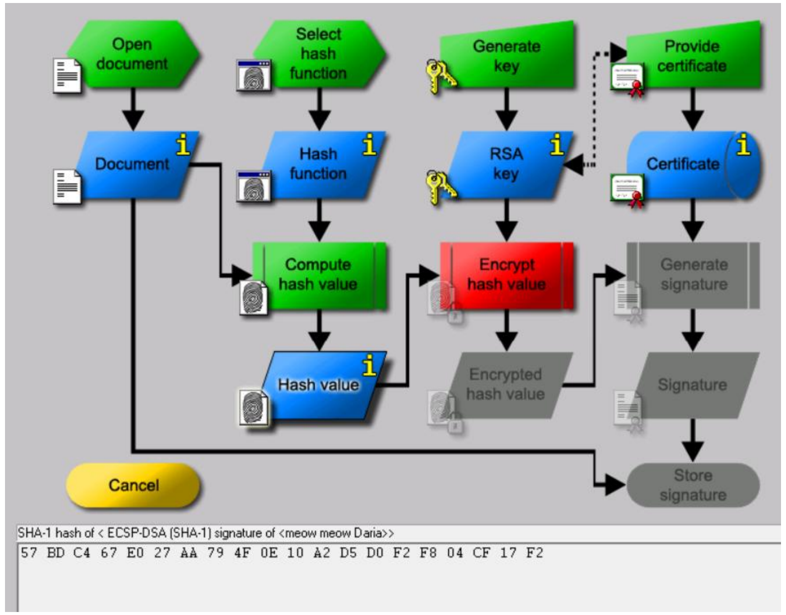




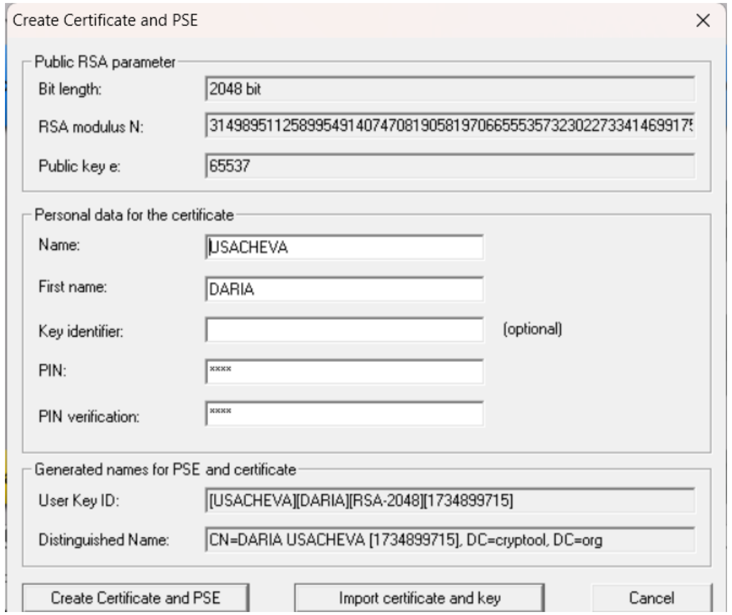
Демонстрация процесса подписи в среде PKI

Демонстрация этапов создания электронной подписи в среде PKI

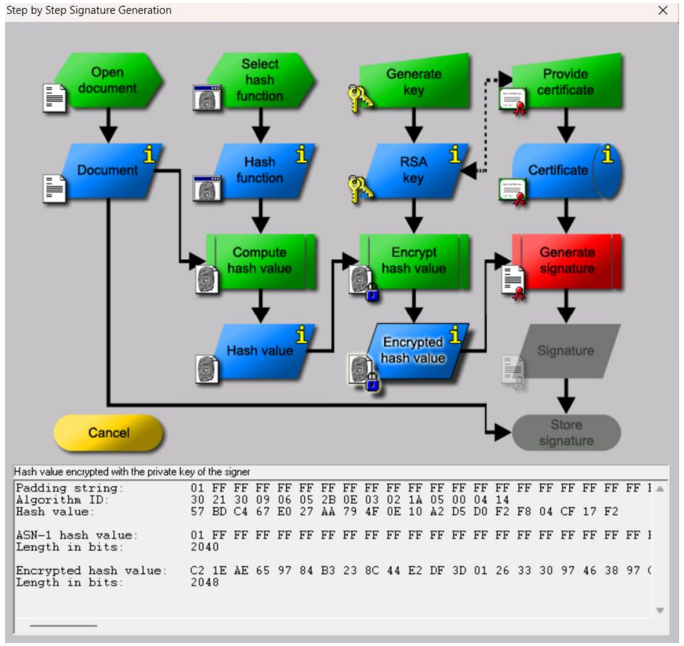
Вычисление дайджеста файла



Импортирование сгенерированной ранее ключевой пары RSA-2048

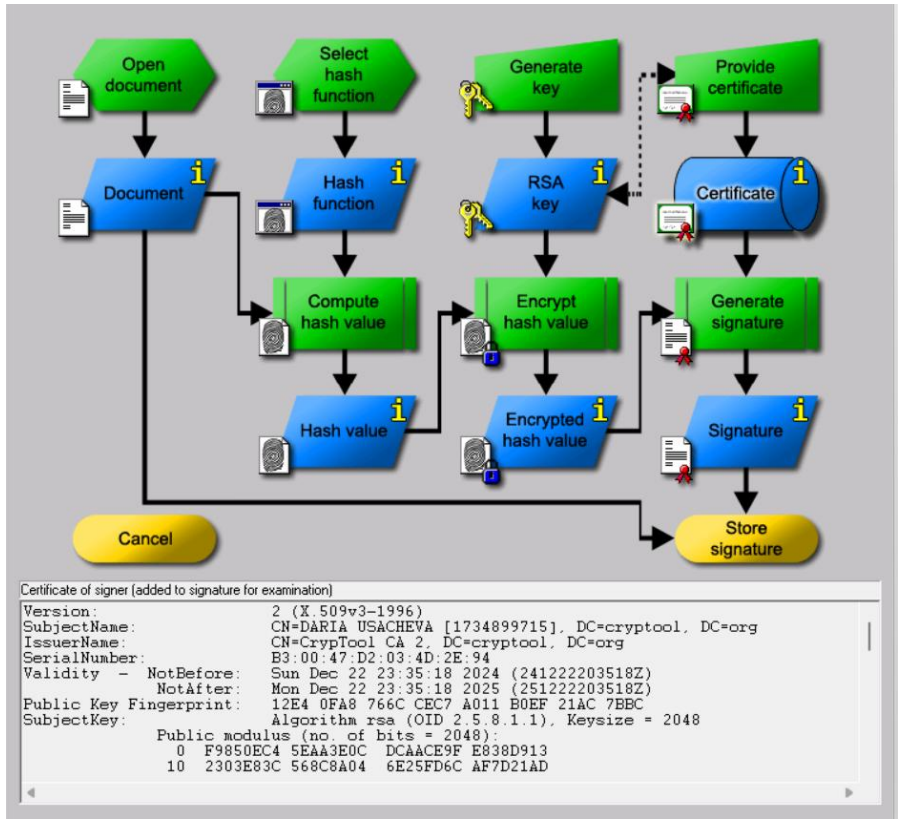


Шифрование ключа функцией диверсификации

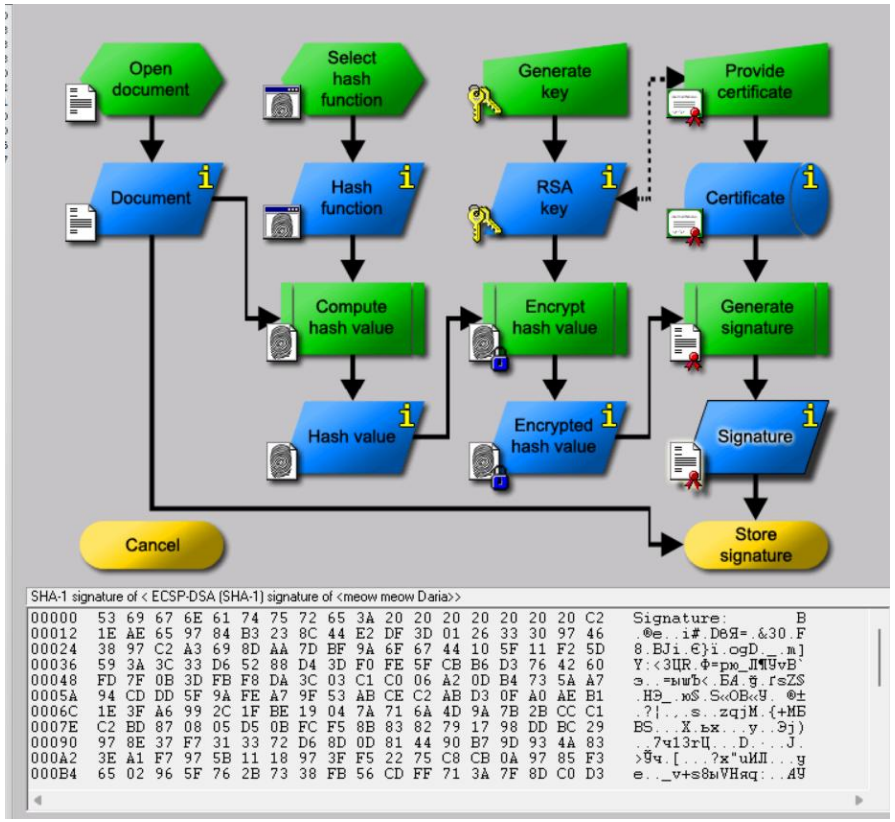


Демонстрация этапов создания электронной подписи в среде PKI

Сертификат



Создание подписи документа



Структура сертификата

Подписание своего отчета

Заключение