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

Лабораторная работа №8

Изучение алгоритмов создания и проверки электронной подписи

Студент: _____ Порошина Алина, группа 0361

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

Санкт-Петербург 2024

Цель работы

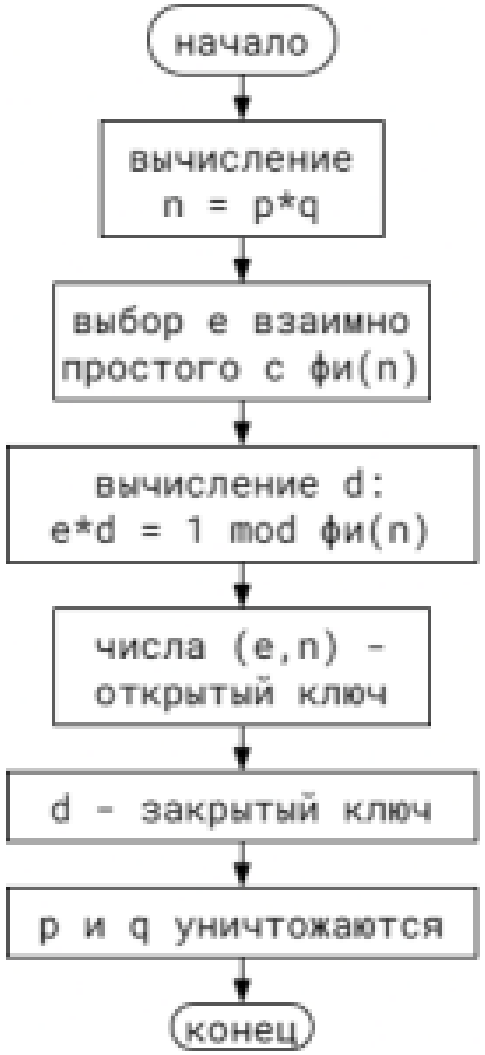
Цель работы:

Приобретение знаний и умений в области алгоритмов создания и проверки электронной подписи.

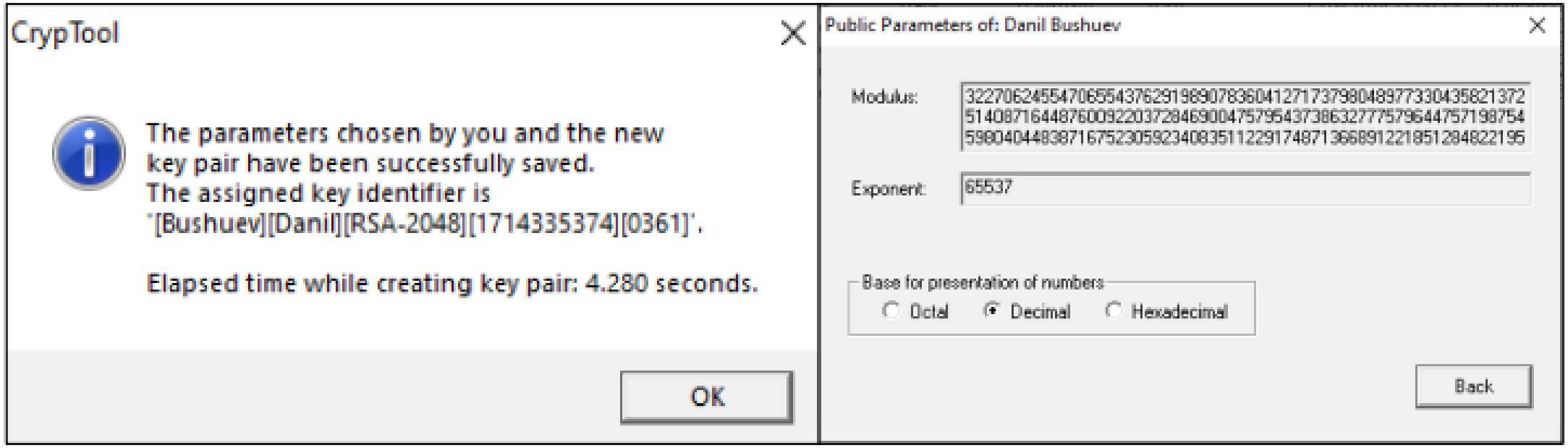
Задачи:

1. Изучить генерацию ключевых пар;
2. Изучить процессы создания и проверки электронных подписей;
3. Изучить создание и проверку электронной подписи на основе эллиптических кривых;
4. Продемонстрировать процесс подписи в среде PKI;
5. Подписать свой отчет

Генерация ключевых пар для алгоритма RSA

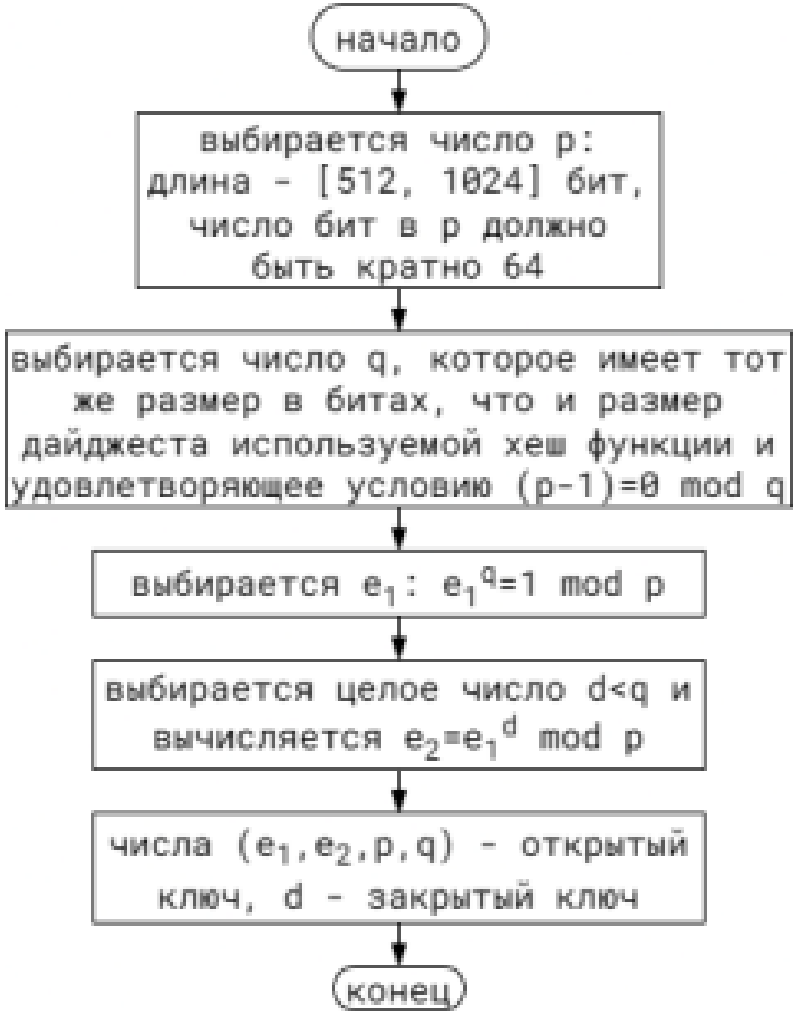


Генерация ключевых пар по алгоритму RSA

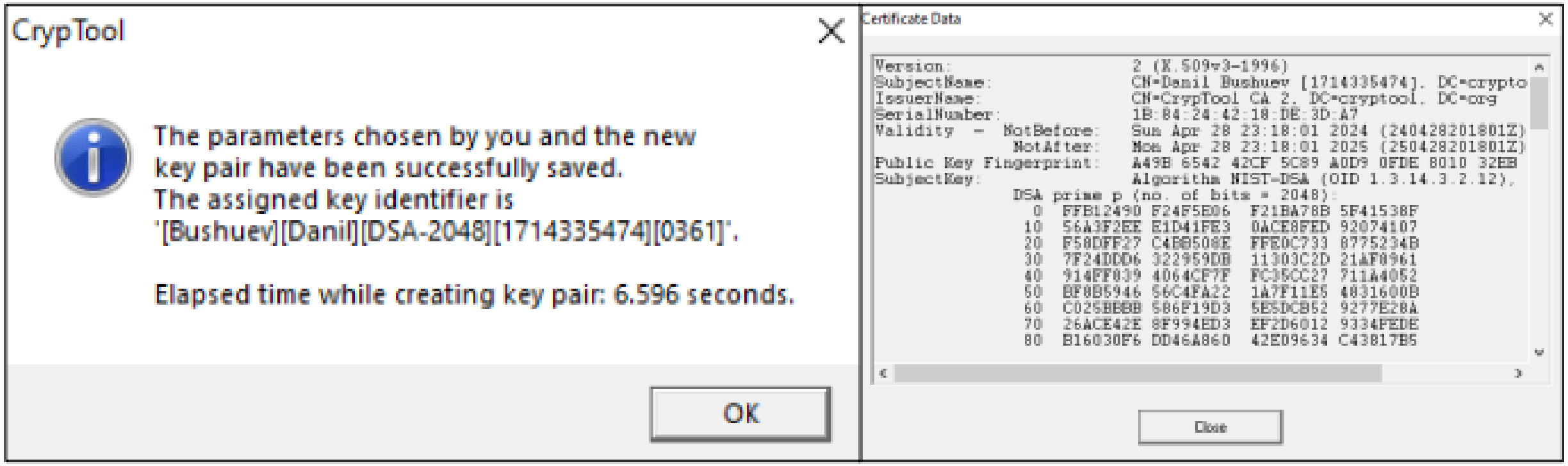


0361
HOLD ON

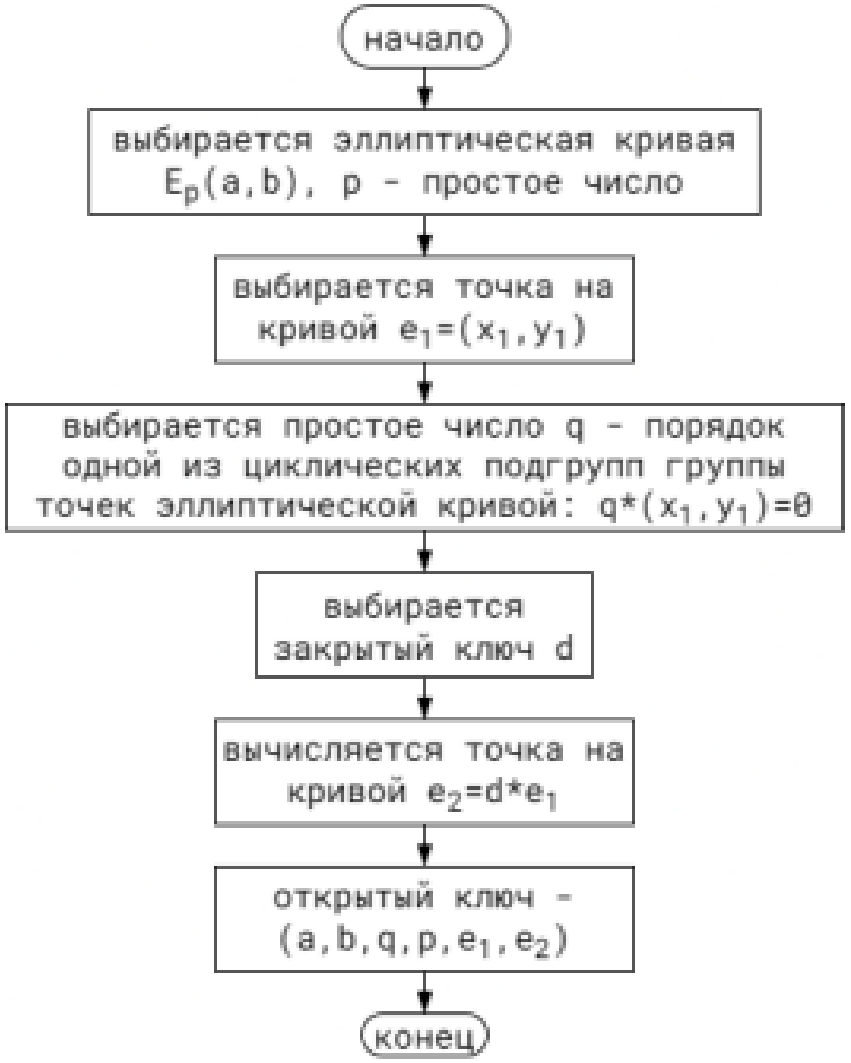
Генерация ключевых пар для алгоритма DSA



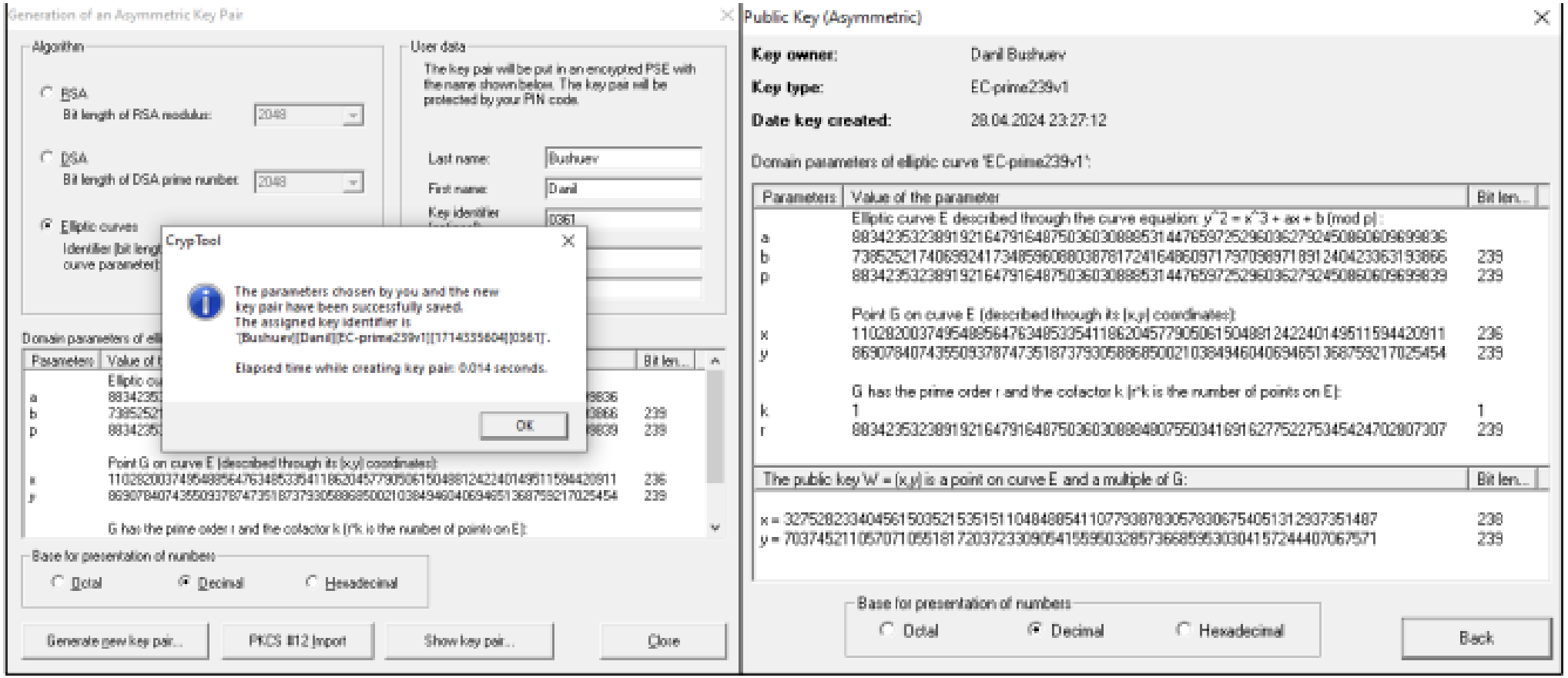
Генерация ключевых пар по алгоритму DSA



Генерация ключевых пар для алгоритма ECDSA



Генерация ключевых пар по алгоритму ECDSA



0361
HOLD ON

Таблица сравнения времени генерации ключей

Алгоритм	Время
RSA-2048	4.280 с
DSA-2048	6.596 с
ECDSA-239	0.014 с

Схема создания и проверки эл. подписи

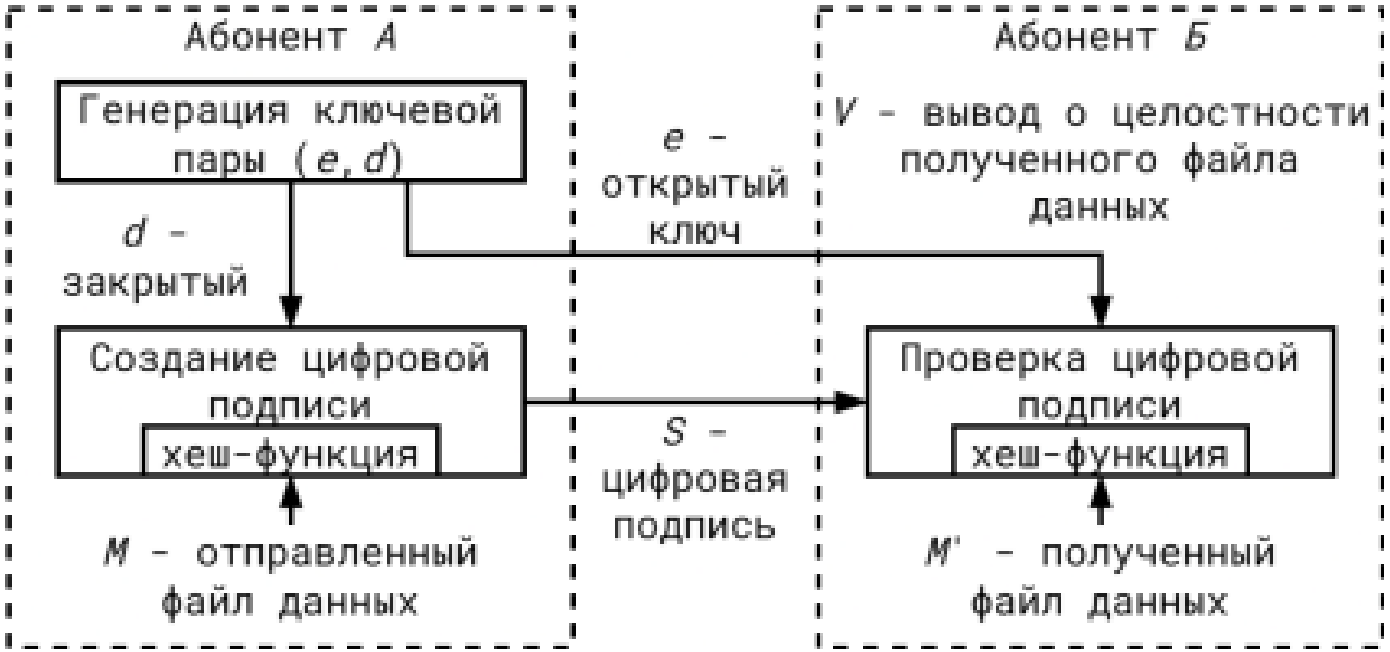
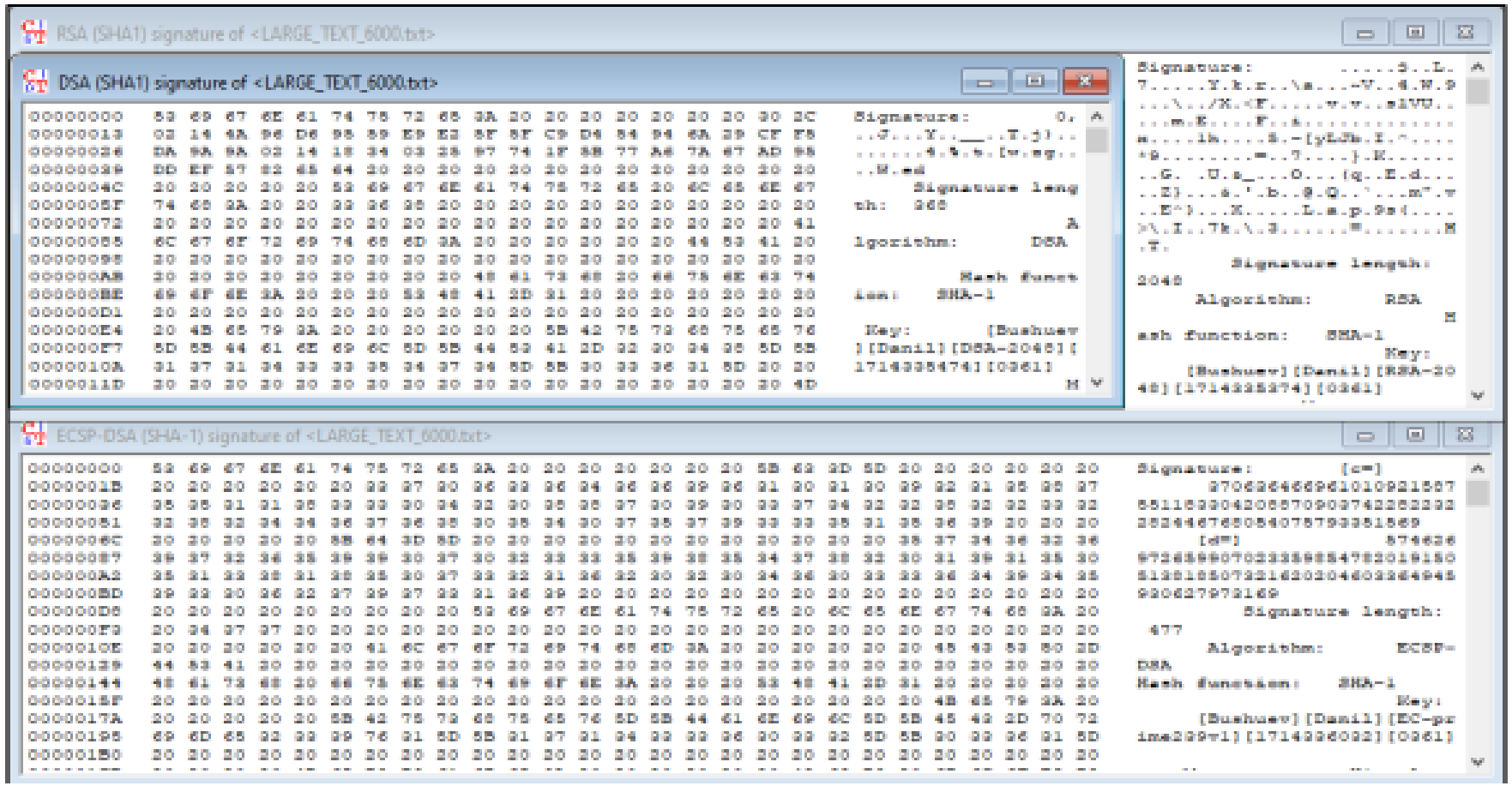


Таблица с временами создания эл. подписи

Алгоритм	Время
RSA-2048	0.000 с
DSA-2048	0.006 с
ECDSA-239	0.002 с

Скриншот со значением эл. подписи



Скриншот с результатами проверки подписи

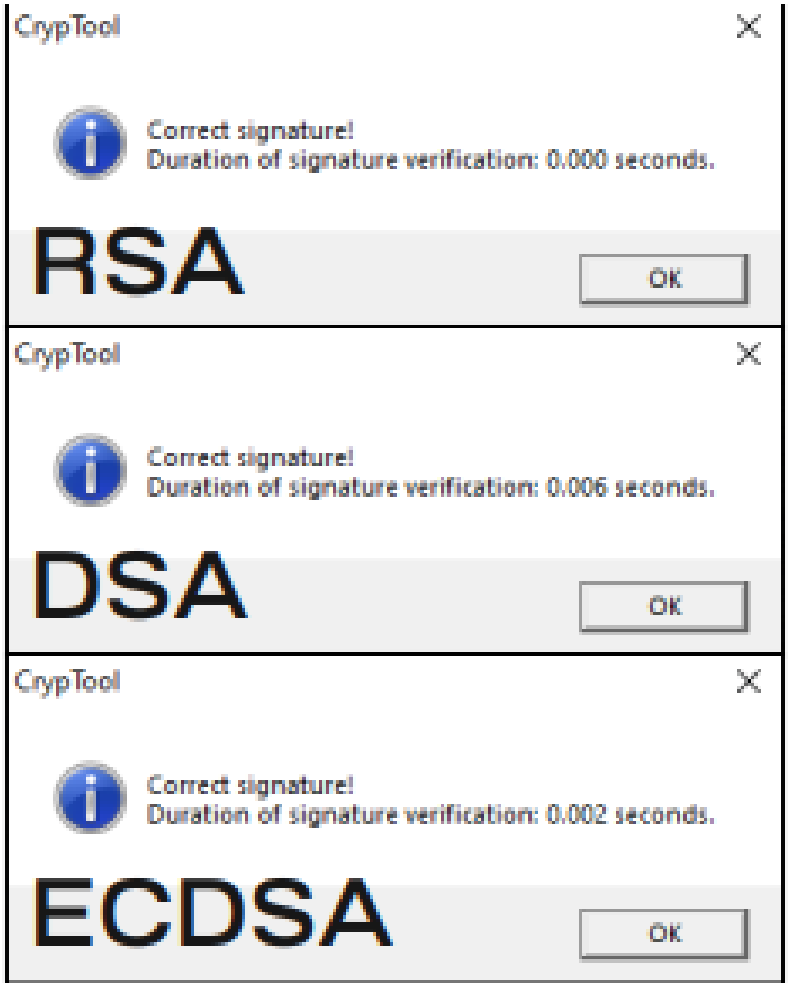
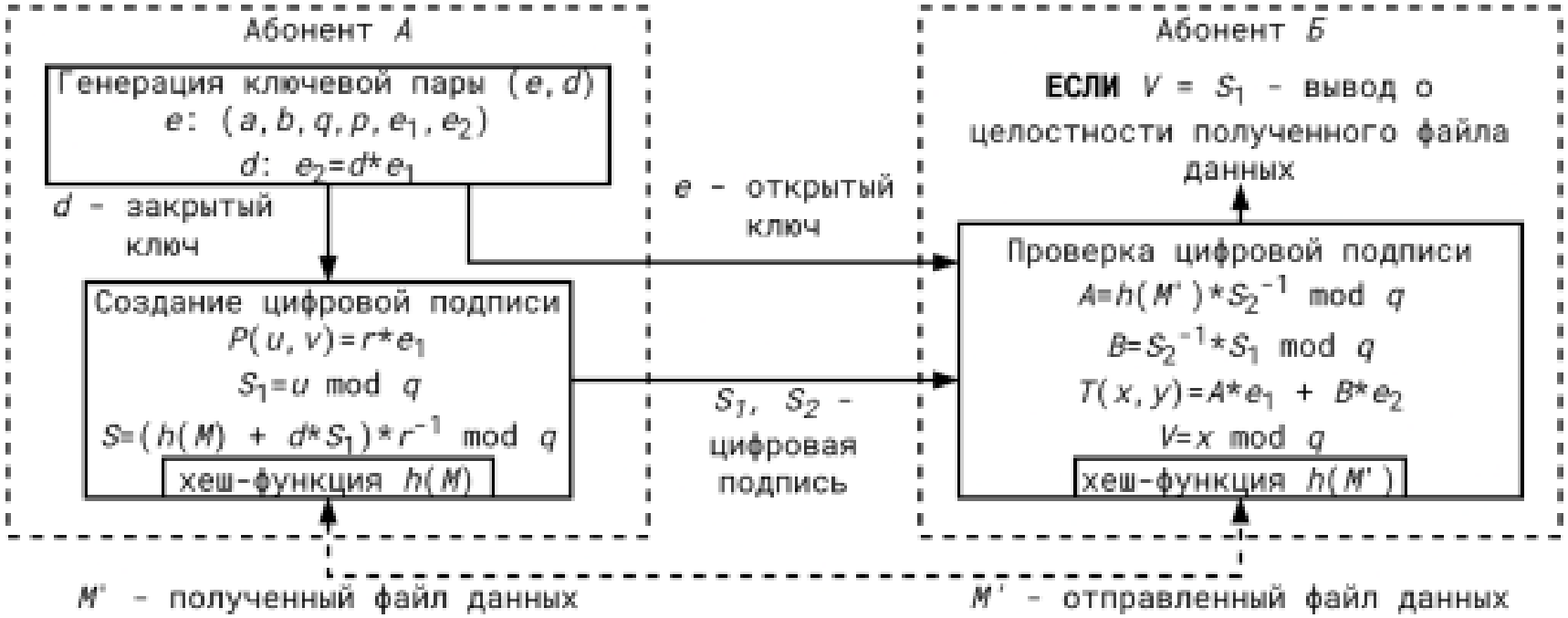


Схема алгоритма формирования и проверки подписи ECDSA



Результаты пошагового выполнения ECDSA

```
Domain parameters to be used 'EC-prime239v1':
  a  = 883423532389192164791648750360308885314476597252960362792458860609699836
  b  = 738525217486992417348596088038781724164860971797098971891248423363193866
  Gx = 110282003749548856476348533541186204577905861504881242240149511594420911
  Gy = 869078487435509378747351873793058868500210384946048694651368759217025454
  k  = 1
  r  = 883423532389192164791648750360308884807550341691627752275345424702807307
Secret key s of the signature originator:
  s  = 707328883901619138220476043702042175766229892760791920085014986025689123
Chosen signature algorithm: ECSP-DSA with hash function SHA-1
Size of message M to be signed: 4442 bytes
```

```
Calculate a 'hash value' f (message representative) from message M, using the chosen hash function SHA-1.
  f  = 3011185580700782549152046876915584615164380336O5
```

```
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 on the elliptic curve):
  u  = 162880155804802734239219351703776350449902639877642352548340786575160 76
  Vx  = 355787363623573152085349425762580638241529349826254670065996400739150134
  Vy  = 722208265230249227501968463299758612601318069705823540091601631728009146
```

```
Convert the group element Vx (x co-ordinates of point V on elliptic curve) to the number i:
  i  = 355787363623573152085349425762580638241529349826254670065996400739150134
```

```
Calculate the number c = i mod r (c not equal to 0):
  c  = 355787363623573152085349425762580638241529349826254670065996400739150134
```

```
Calculate the number d = u^(-1)*(f + s*c) mod r (d not equal to 0):
  d  = 297928393703928349119625043027634971542313049710796143077973238124297694
```

Проверка подписи ECSP-DSA

Проверка лекционного материала

Генерация ключей ECDSA

Выбирается эллиптическая кривая	E7(3, 5)
Выбирается точка на кривой	e1=(4, 5)
Подбирается порядок циклической подгруппы	q=7
Выбирается целое число и назначается закрытым ключом	d=4
Вычисляется другая точка на кривой	e2=(6,1)
Объявляется открытый ключ	(3, 5, 7, 7, (4, 5), (6, 1))

Проверка лекционного материала

ECDSA подписание

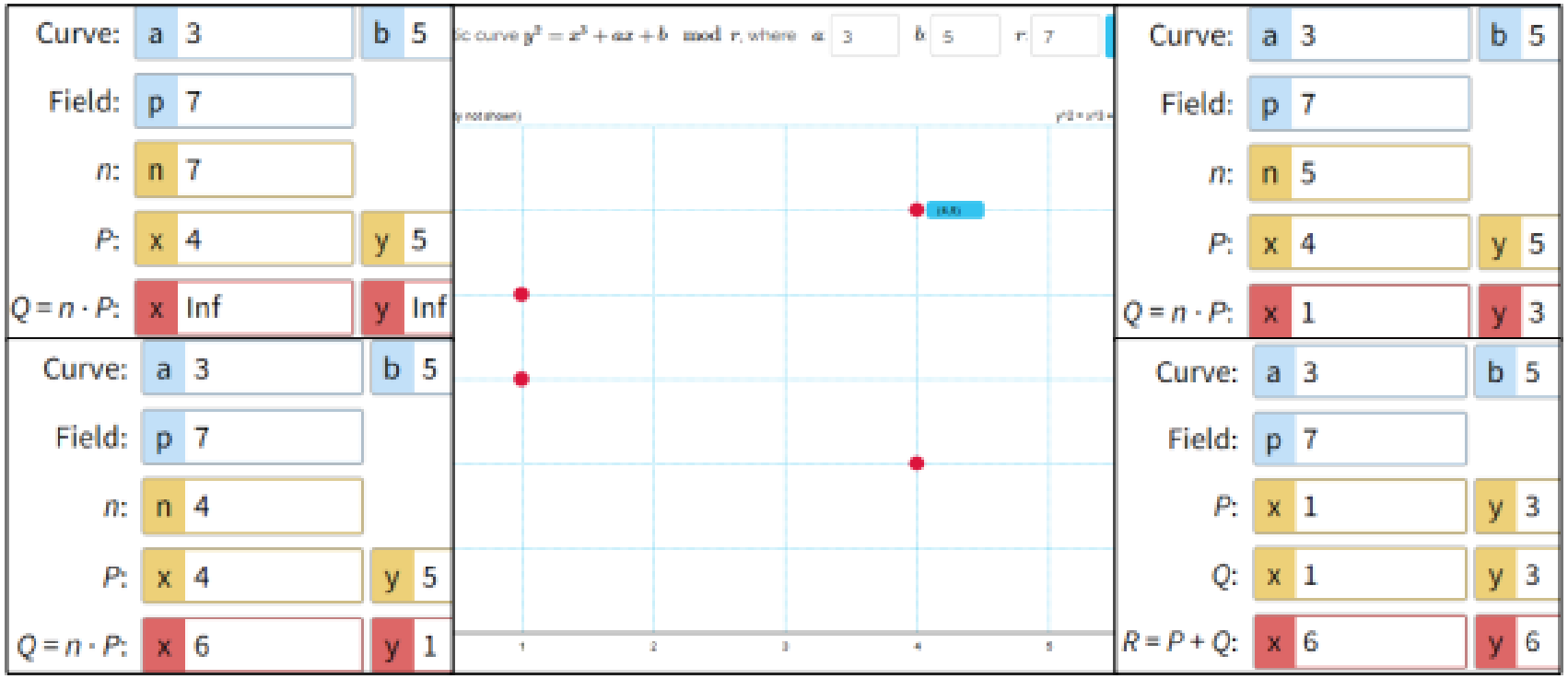
Выбирается секретное случайное число	r=3
Выбирается третья точка на кривой	e3=(6, 6)
Используем абсциссу, чтобы вычислить 1 часть подписи	S1=6
Вычисляем дайджест	h(M)=10
Используем дайджест сообщения h(M), закрытый ключ d, секретное случайное число r и S1, чтобы вычислить 2 часть подписи	S2=(10+4*6)*3^(-1) mod 7 S2=2

Проверка лекционного материала

ECDSA проверка

Используем M, S1, S2 для получения промежуточных результатов A и B	A=10*2^(-1) mod 7 B=2^(-1)*6 mod 7 A=5 B=3
Восстанавливаем третью точку	T=5*(4,5)+3*(6,1) T=(1,3)+(1,3) T=(6,6)
Верификатор V=x mod q сравниваем с S1	V=6 mod 7 V=6 V=S

Проверка лекционного материала



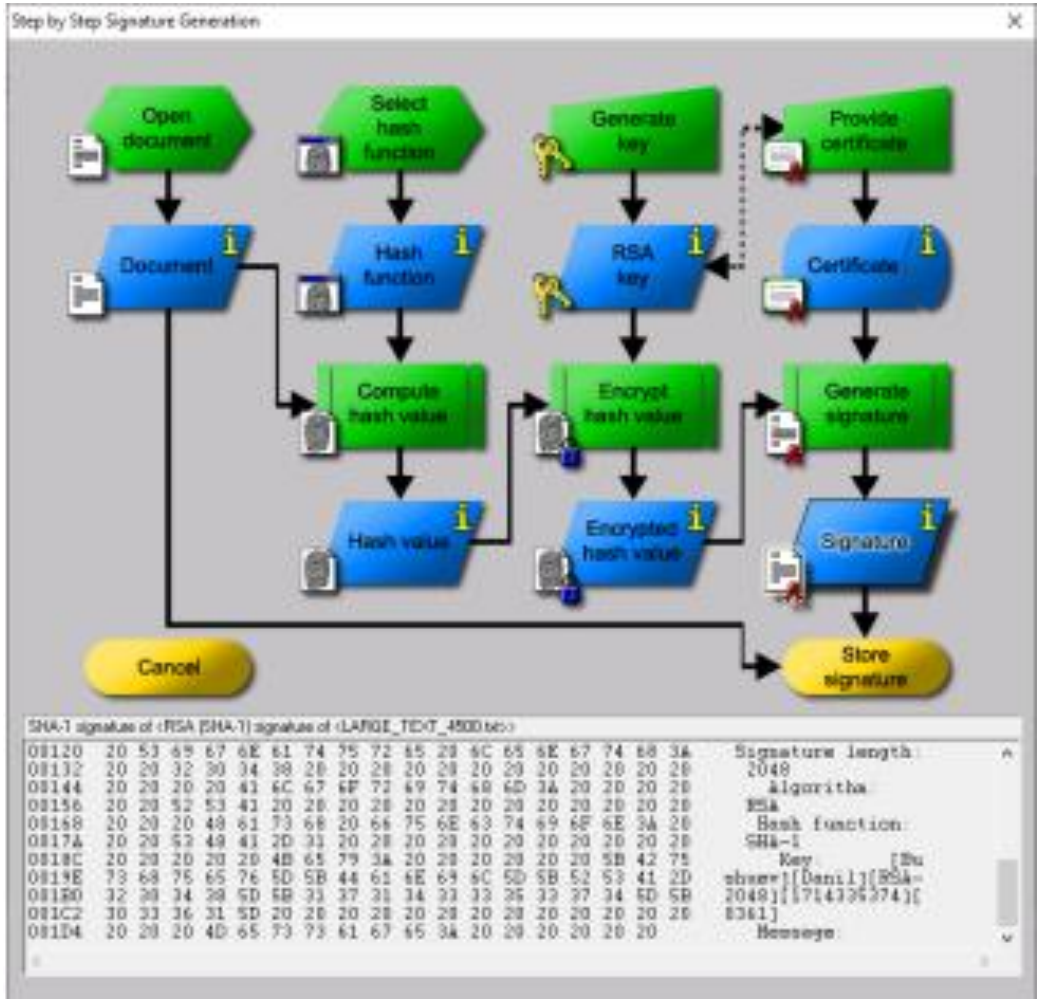
Описание структуры сертификата (CrypTool 1)

```
Version: 2 (X.509v3-1996)
SubjectName: CN=Danil Bushuev [1714335374], DC=cryptool, DC=org
IssuerName: CN=CrypTool CA 2, DC=cryptool, DC=org
SerialNumber: 02:93:82:72:A5:E5:2B:99
Validity - NotBefore: Sun Apr 28 23:16:18 2024 (240428201618Z)
           NotAfter: Mon Apr 28 23:16:18 2025 (250428201618Z)
Public Key Fingerprint: AE33 205D C83F 21D5 9B66 E4F3 DE09 223D
SubjectKey: Algorithm RSA (OID 2.5.8.1.1), Keysize = 2048
Public modulus (no. of bits = 2048):
      0  FFA1F143 11BB7E7C  BA823753 A82F61FA
     10  B5470720 1731AB24  270BBF79 B350215A
     20  B96166BF 1A728A1B  0306B2B0 84400834
     30  3F3AEF1D 16321F15  BC1FDF5C E97736B5
     40  94A82821 F1669D42  6B1731FF 28AB8FD0
     50  E3565457 4D51D2EC  3D1CEDD3 C49E3000
     60  C94D2DE2 BDE6D41D  750B325E B1CCB465
     70  C2BB5456 81EF002C  79FE036C 7286B79B
     80  956B78F0 650D9ABF  23B7199F FD5AB017
     90  E14A128D EEE8B20F  634DAEF7 F77A706C
     A0  AEDF86C9 F7DB91B1  47871100 ECAF8D58
     B0  0B16CCE9 76A7134D  D0067E49 30C7CC7A
     C0  77493552 EF0DAD0D  C4E81DBF 2D2F239F
     D0  74516B09 BC94AD7F  D95E5646 0950B926
     E0  7A87CA06 F08FE14C  0C74440B 85811CEC
     F0  2A29FDF5 E032F41D  30A389AE 3FD1617B
Public exponent  (no. of bits = 17): 0  010001
```

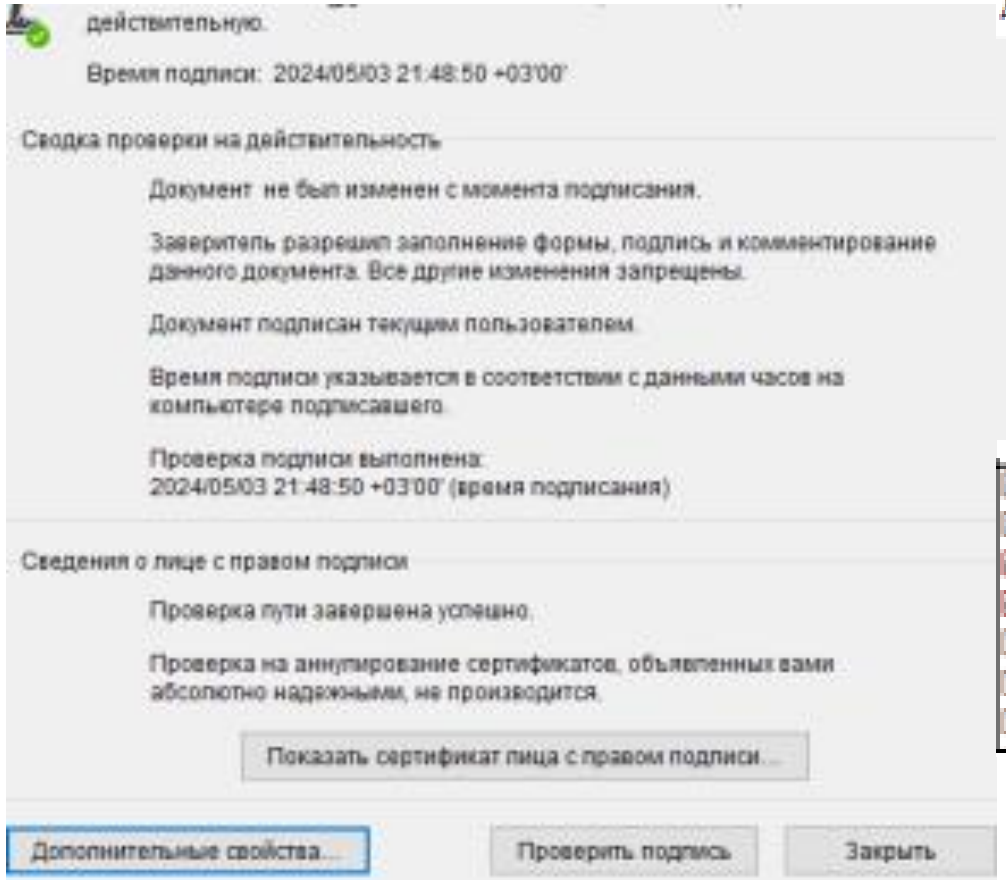
```
Certificate extensions:
Private extensions:
 OID 2.206.5.4.3.2:
  PrintableString: [Bushuev][Danil][RSA-2048][1714335374][0361]
Signature:
Algorithm sha1WithRSASignature (OID 1.3.14.3.2.29), NULL
      0  9ED78932 2B5D61A1  9276C77C 5CB36984
     10  F73C3BBB 5156297E  679CC48C 51BB3653
     20  4FC7CE5B D1EE428B  6A74E42E AAF9EE19
     30  20C98095 5443850A  CB105435 D15841E7
     40  E555C380 7A362E96  C7980952 62AF8B6E
     50  71999B7E C707E2E0  A134E4DF 1FF05C6C
     60  6D4016EE 3D0E6859  68404C98 5A9E3330
     70  B3FF3EFD 2B9D1CAC  ED31FB9B B8517AD2
     80  4542BB08 AC4F106A  2345BD75 A6AFC754
     90  642FEAE2 737EF10E  28444B7C 4DAEEFD2
     A0  986F8CBA E57D2E2F  348882C1 836D938C
     B0  FEDCB968 5221CAF9  64711036 7287B50A
     C0  4E729273 F76CC994  0BEAC2DD 8DC96297
     D0  3667FB88 D3641666  BF5599DF 2CEEC4DF
     E0  98E9AC8C CE56FEBE  30578FE0 57EE4BA5
     F0  C7D36B29 B3509FEE  6F934962 5A0E3201
Certificate Fingerprint (MD5):
E1:5B:5E:C8:02:73:73:D6:D0:35:51:F6:25:42:AD:C1
Certificate Fingerprint (SHA-1):
CF7C 3F6C ACCE F142 CC35 1C5B 17F8 951A C837 1CA8
```

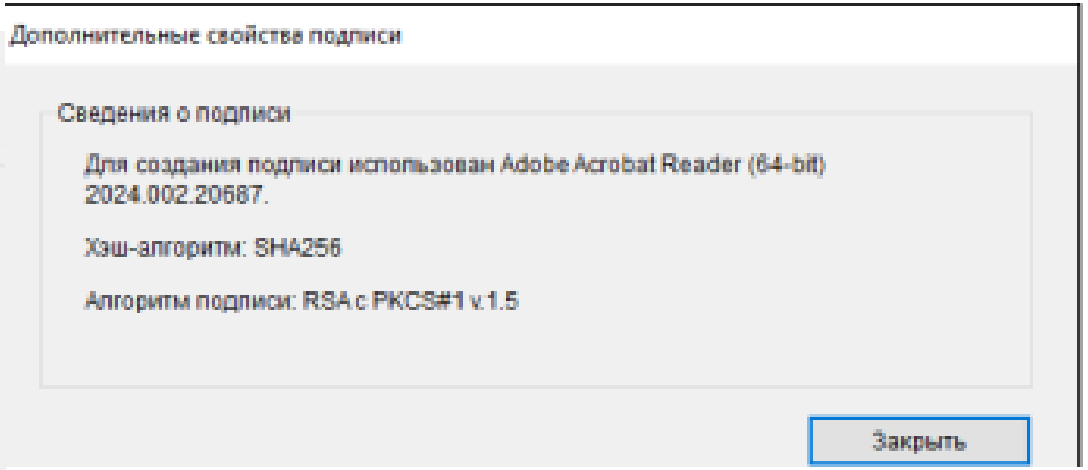
0361
HOLD ON

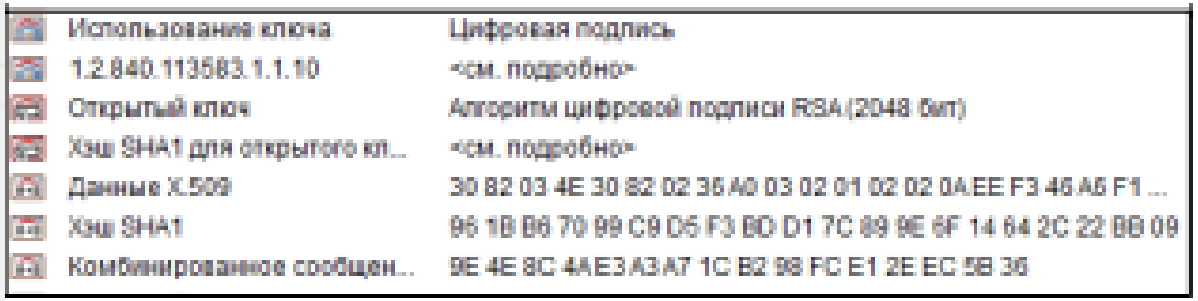
Схема процедуры подписания



Свойства подписи и сертификата

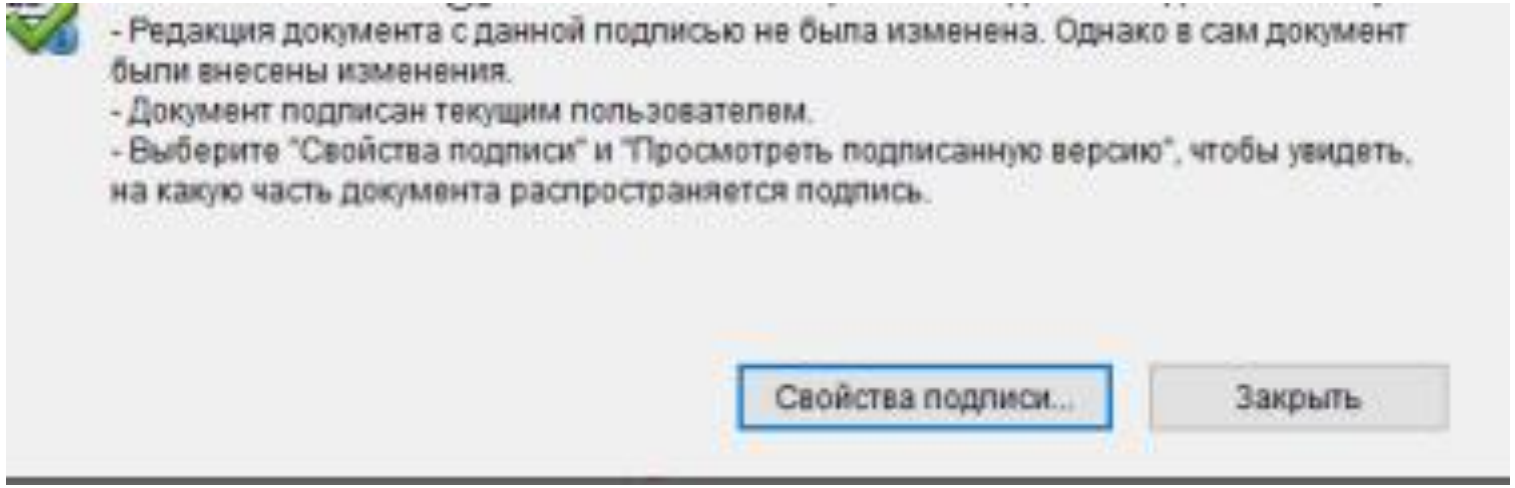






0361
HOLD ON

Результат проверки после изменений



Выводы

1. Были сгенерированы ключевые пары для алгоритмов RSA-2048, DSA2048 и EC-239. Проведено сравнение времени генерации ключа. Наилучший показатель у алгоритма EC-239.
2. Был подписан текстовый файл размером 4500 символов и проведена проверка подписи. Проведено сравнение времени для каждого из трех алгоритмов. Наилучший показатель у алгоритма EC-239. Документ подписанный этим алгоритмом был проверен после модификации, она была обнаружена.
3. Был исследован протокол электронной подписи ECSP-DSA в пошаговом режиме. С помощью данного протокола создана и проверена цифровая подпись. Был проверен лекционный материал.
4. Был создан сертификат для полученного ключа RSA-2048 и с помощью него была создана электронная подпись.
5. Был подписан отчет и проведена проверка после изменения файла.

Спасибо за внимание!
Готова ответить на ваши вопросы.