

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №8
по дисциплине «Криптография и защита информации»
Тема: Изучение цифровой подписи

Студент гр. 8383

Ларин А.

Преподаватель

Племянников А.К.

Санкт-Петербург

2021

Цель работы

Исследовать алгоритмы создания и проверки цифровой подписи, алгоритмы генерации ключевых пар для алгоритмов цифровой подписи RSA, DSA, ECDSA и получить практические навыки работы с ними, в том числе с использованием приложения Cryptool 1 и 2.

1. Генераторов ключевых пар

Генерация ключевых пар для алгоритма RSA

Генерация двух больших простых чисел p и q (p и q держаться в секрете).

1. Вычисление $n = p * q$
2. Выбор произвольного e ($e < n$), взаимно простого с $\phi(n)$.
3. Вычисление d : $e * d = 1 \bmod \phi(n)$.
4. Числа (e, n) – открытый ключ, d – закрытый ключ, p и q уничтожаются.

Генерация ключевых пар для алгоритма DSA

1. Выбирается число p : длина - $[512, 1024]$ битов и число битов в p должно быть кратно 64.
2. Выбирается число q , которое имеет тот же самый размер дайджеста 160 битов, такое, что: $(p - 1) = 0 \bmod q$.
3. Выбирается $e_1: e_1^q = 1 \bmod p$.
4. Выбирается целое число $d < q$ и вычисляется $e_2 = e_1^d \bmod p$.
5. Числа (e_1, e_2, p, q) – открытый ключ, d – закрытый ключ.

Генерация ключевых пар для алгоритма ECDSA

1. Выбирается эллиптическая кривая $E_p(a, b)$, p – простое число.
2. Выбирается точка на кривой $e_1 = (x_1, y_1)$
3. Выбирается простое число q – порядок одной из циклических подгрупп группы точек эллиптической кривой: $q \times (x_1, y_1) = O$.
4. Выбирается закрытый ключ d .
5. Вычисляется точка на кривой $e_2 = d \times e_1$.
6. Открытый ключ - (a, b, q, p, e_1, e_2) .

Задание

1. Перейти к утилите «Digital Signatures/PKI->PKI/Generate...».
2. Сгенерировать ключевые пары по алгоритмам RSA-2048, DSA-2048, EC-239. Зафиксируйте время генерации в таблице.
3. С помощью утилиты «Digital Signatures/PKI-> PKI/Display...» вывести сгенерированный открытый ключ и сохранить соответствующий скриншот.

Выполнение

1. Перешли

Generation of Asymmetric Key Pair

Algorithm:

- ☐ RSA
Bit length of RSA modulus: 2048
- ☐ DSA
Bit length of DSA prime number: 2048
- ☒ Elliptic curves
Identifier (bit length and curve parameter): prime239v1

User data:

The key pair will be put in an encrypted PSE with the name shown below. The key pair will be protected by your PIN code.

Last name: asd1
First name: asd1
Key identifier (optional):
PIN:
PIN verification:

Domain parameters of elliptic curve 'prime239v1':

Parameters	Value of the parameter	Bit len...
Elliptic curve E described through the curve equation: $y^2 = x^3 + ax + b \pmod{p}$:		
a	883423532389192164791648750360308885314476597252960362792450860609699836	239
b	738525217406992417348596088038781724164860971797098971891240423363193866	239
p	883423532389192164791648750360308885314476597252960362792450860609699839	239
Point G on curve E (described through its (x,y) coordinates):		
x	110282003749548856476348533541186204577905061504881242240149511594420911	236
y	869078407435509378747351873793058868500210384946040694651368759217025454	239
G has the prime order r and the cofactor k (r*k is the number of points on E):		

Base for presentation of numbers:

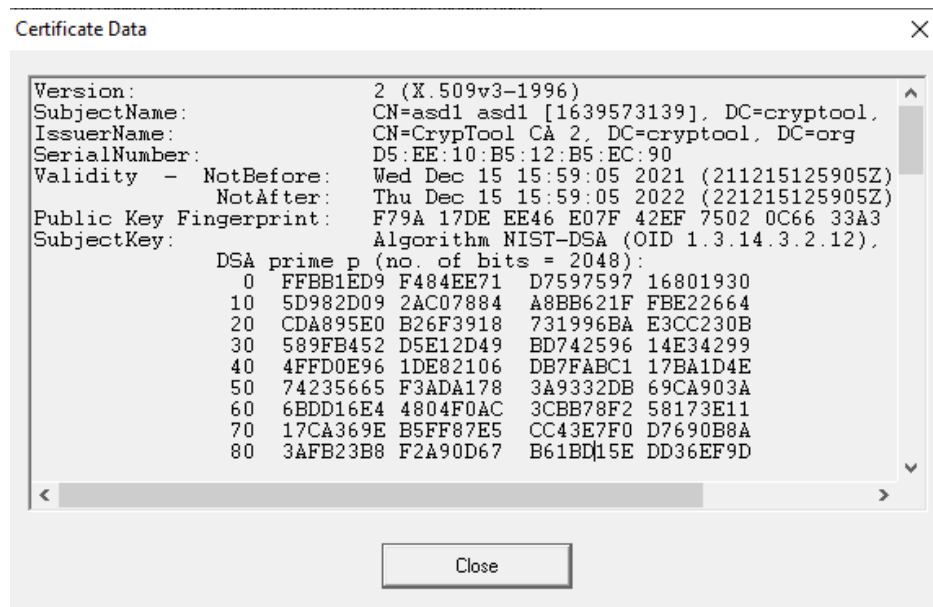
- ☐ Octal
- ☒ Decimal
- ☐ Hexadecimal

Generate new key pair... PKCS #12 Import Show key pair... Close

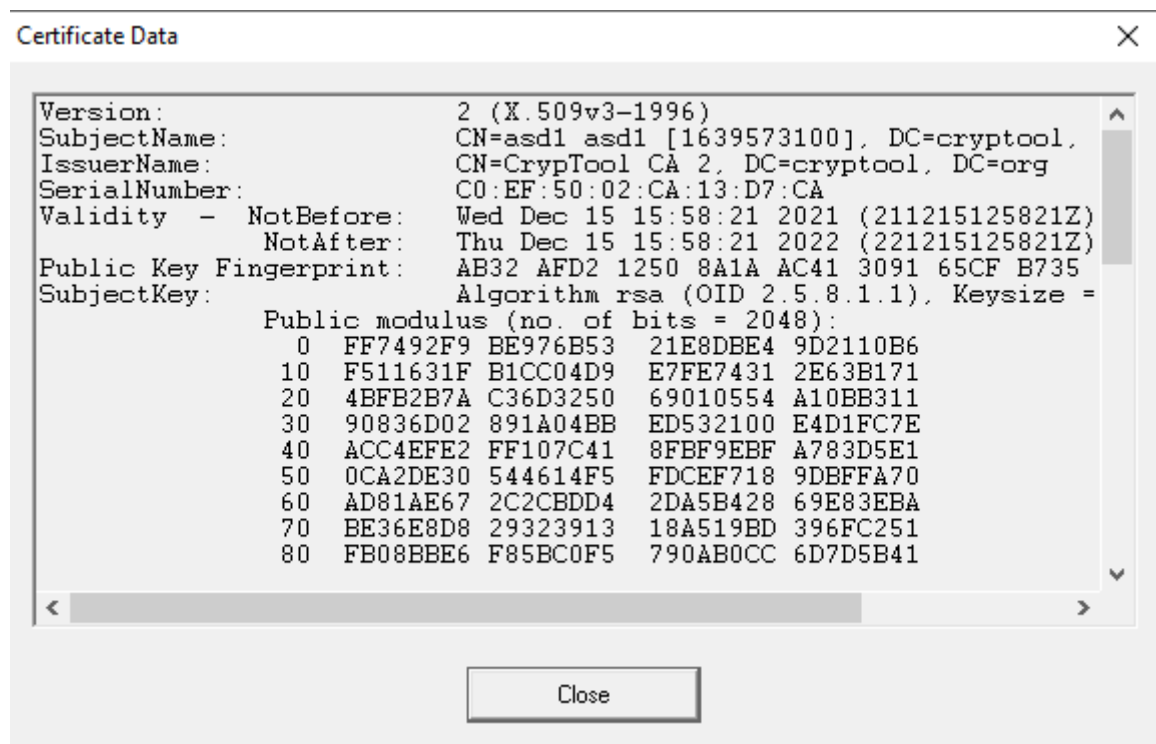
2. Сгенерированы ключи методами . Время генерации в таблице

RSA-2048	DSA-2048	EC-239
0.937	6.638	0.033

3. DSA-2048



4. RSA-2048



5. EC-239

Key owner: asd1 asd1
Key type: EC-prime239v1
Date key created: 15.12.2021 15:59:38

Domain parameters of elliptic curve 'EC-prime239v1':

Parameters	Value of the parameter	Bit len...
Elliptic curve E described through the curve equation: $y^2 = x^3 + ax + b \pmod{p}$:		
a	883423532389192164791648750360308885314476597252960362792450860609699836	239
b	738525217406992417348596088038781724164860971797098971891240423363193866	239
p	883423532389192164791648750360308885314476597252960362792450860609699839	239
Point G on curve E (described through its (x,y) coordinates):		
x	110282003749548856476348533541186204577905061504881242240149511594420911	236
y	869078407435509378747351873793058868500210384946040694651368759217025454	239
G has the prime order r and the cofactor k ($r \cdot k$ is the number of points on E):		
k	1	1
r	883423532389192164791648750360308884807550341691627752275345424702807307	239
The public key $W = (x,y)$ is a point on curve E and a multiple of G:		
x	298036930856746846668885329515860019204999767607561119387347995029673264	238
y	615485127060011310837218439858344118268304193611761418218287594061455287	239
The secret key s is the solution of the EC discrete log problem $W = x \cdot G$ (x unknown):		
s	759642526515539634409328795580125896284304311498653288552379937750093587	239

Base for presentation of numbers:

☐ Octal ☒ Decimal ☐ Hexadecimal

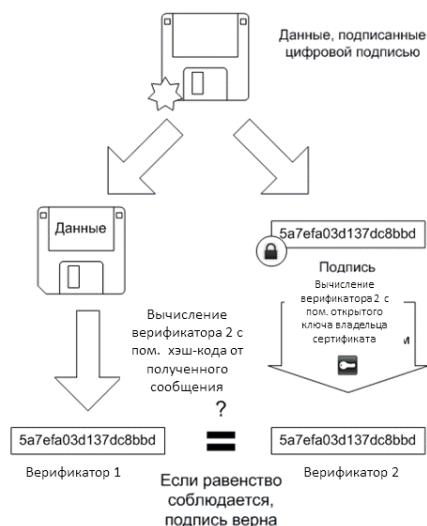
Back

2. Процессы создания и проверки цифровой подписи

Создание цифровой подписи



Проверка цифровой подписи



Задание

1. Открыть текст не менее 5000 знаков. Перейти к приложению Digital Signatures/PKI-> Sign Document...

2. Задайте хэш-функцию, и другие параметры цифровой подписи.
3. Создайте подпись ключами, сгенерированными в предыдущем задании. Зафиксируйте время создания цифровой подписи для каждого ключа.
4. Сохраните скриншот цифровой подписи с помощью приложения Digital Signatures/PKI-> Extract Signature.
5. Выполните процедуру проверки подписи Digital Signatures/PKI→ Verify Signature для случаев сохранения и нарушения целостности исходного текста. Сохраните скриншоты результатов.

Выполнение

1. Взял полный текст поэмы Raven.

Sign a Document ✕

Choose hash function

Algorithm:	Output length
<input type="radio"/> MD2	128 bits
<input checked="" type="radio"/> MD5	128 bits
<input type="radio"/> RIPEMD-160	160 bits
<input type="radio"/> SHA	160 bits
<input type="radio"/> SHA-1	160 bits

Choose signature algorithm

Factorization based algorithms

☒ RSA

Discrete logarithm based algorithms

☐ DSA

Elliptic curve based algorithms

☐ ECSP-DSA

☐ ECSP-NR

Presentation format

☐ Affine coordinates

☒ Projective coordinates

Choose a key/PSE to be used when signing

Last name	First name	Key type	Key identifier	Created	Internal ID no.
asd1	asd1	DSA-2048		15.12.2021 15:58:59	1639573139
asd1	asd1	EC-prime239v1		15.12.2021 15:59:38	1639573178
asd	asd	RSA-2048		15.12.2021 15:56:27	1639572987
HybridEncrypti...	Bob	EC-prime239v1	PIN=1234	09.05.2007 12:21:14	1178702474
larin	anton	RSA-1024		09.12.2021 18:15:18	1639062918
larin	anton	RSA-2048		09.12.2021 18:16:07	1639062967
larin	anton	RSA-512		09.12.2021 18:12:57	1639062777
larin	anton	RSA-768		09.12.2021 18:14:44	1639062884
SideChannelAt...	Bob	RSA-512	PIN=1234	06.07.2006 12:51:34	1152179494

Listed key types:

☒ RSA keys

☒ DSA keys

☒ EC keys

PIN code for chosen PSE:

☐ Display signature time

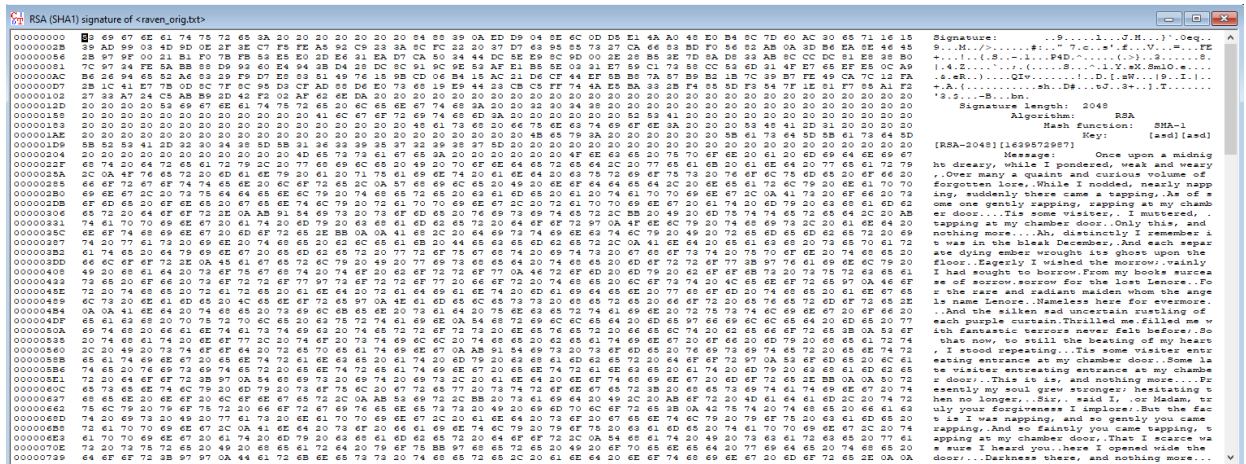
☐ Display intermediate results

Sign

Cancel

2. Задана функция SHA-1
- 3.

RSA-2048	DSA-2048	EC-239
0.014	0	0



4. Извлечённая подпись

Extracted Signature

Signer: asd1 asd1

Used key: EC-prime239v1; created 15.12.2021 15:59:38

Signature algorithm: ECSP-DSA with hash function SHA-1

Signature:

```
c = 50659195318305430192510572725298591245112396003992703400460556658518683
d = 74626393287663206754084470532739666216485505498821909873767001698925516'
```

Length of signature: 478 bits

Options for presentation of signature

Numbers: ☐ Octal ☒ Decimal ☐ Hexadecimal

Hex dump (hexadecimal and ASCII): ☐

Signed message:

```
00000 4F 6E 63 65 20 75 70 6F 6E 20 61 20 6D 69 Once upon a mi
0000E 64 6E 69 67 68 74 20 64 72 65 61 72 79 2C dnight dreary,
0001C 20 77 68 69 6C 65 20 49 20 70 6F 6E 64 65 while I ponde
0002A 72 65 64 2C 20 77 65 61 6B 20 61 6E 64 20 red, weak and
00038 77 65 61 72 79 2C 0A 4F 76 65 72 20 6D 61 weary, .Over ma
00046 6E 79 20 61 20 71 75 61 69 6E 74 20 61 6E ny a quaint an
00054 64 20 63 75 72 69 6F 75 73 20 76 6F 6C 75 d curious volu
00062 6D 65 20 6F 66 20 6F 72 67 6F 74 74 65 me of forgotte
```

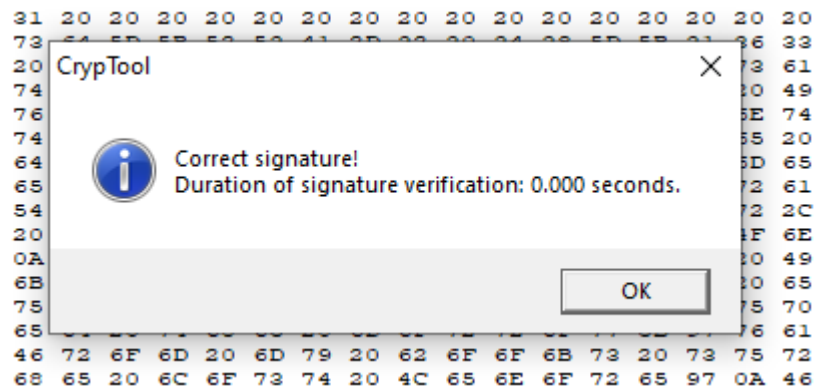
Length of message: 6245 bytes

Verify

Close

5. Результат проверки подписи

Проверка без изменений



Проверка после фальсификации



3. Схемы цифровой подписи на эллиптических кривых

Задание

1. Выполните процедуру создание подписи «Digital Signatures/PKI→ Sign Document...» алгоритмом ECSP-DSA в пошаговом режиме (Display inter. results=ON). Зафиксируйте скриншоты последовательности шагов.
2. Выполните процедуру проверки подписи ECSP-DSA для случаев сохранения и нарушения целостности исходного текста. Сохраните скриншоты результатов.
3. Проверить лекционный материал по ECDSA, выполнив создание и проверку подписи сообщения M (принять $M=h(M)$) приложением Indiv.Procedures->Number Theory...->Point Addition on EC.

Выполнение

1. Начало работы:

Sign a Document

Choose hash function

Algorithm:	Output length
<input type="radio"/> MD2	128 bits
<input type="radio"/> MD5	128 bits
<input type="radio"/> RIPEMD-160	160 bits
<input type="radio"/> SHA	160 bits
<input checked="" type="radio"/> SHA-1	160 bits

Choose signature algorithm

Factorization based algorithms

☐ RSA

Discrete logarithm based algorithms

☐ DSA

Elliptic curve based algorithms

☒ ECSP-DSA

☐ ECSP-NR

Choose a key/

Last name	Internal ID no.
asd1	639573178
HybridEncrypt	78702474

Listed key types:

☐ RSA keys

☐ DSA keys

☒ EC keys

PIN code for chosen PSE:


☒ Display signature time

☒ Display intermediate results

Sign

Cancel

CrypTool

 The calculation "on" the elliptic curve is performed during presentation of the signature intermediate results in affine coordinates. It is not possible to output the intermediate results in projective co-ordinates (x, y, z co-ordinates), i.e. points on an elliptic curve will be output as x, y co-ordinates (affine co-ordinates).

OK

Шаг 0:

Signature Generation - Step By Step

Message M to be signed:

00000	4F 6E 63 65 20 75 70 6F 6E 20 61 20 6D 69	Once upon a mi
0000E	64 6E 69 67 68 74 20 64 72 65 61 72 79 2C	dnight dreary,
0001C	20 77 68 69 6C 65 20 49 20 70 6F 6E 64 65	while I ponde
0002A	72 65 64 2C 20 77 65 61 6B 20 61 6E 64 20	red, weak and
00038	77 65 61 72 79 2C 0A 4F 76 65 72 20 6D 61	weary. Over ma
00046	6E 79 20 61 20 71 75 61 69 6E 74 20 61 6E	ny a quaint an
00054	64 20 63 75 72 69 6F 75 73 20 76 6F 6C 75	d curious volu
00062	6D 65 20 6F 66 20 66 6F 72 67 6F 74 74 65	me of forgotte

Step-by-step signature generation:

Signature originator: asd1 asd1

Domain parameters to be used 'EC-prime239v1':

a = 88342353238919216479164875036030888531447659725296036279245081

b = 7385252174069924173485960880387817241648609717970989718912404:

Gx = 1102820037495488564763485335411862045779050615048812422401495:

Gy = 8690784074355093787473518737930588685002103849460406946513687!

k = 1

r = 8834235323891921647916487503603088848075503416916277522753454:

Secret key s of the signature originator:

s = 7596425265155396344093287955801258962843043114986532885523799:

Step 0 out of a maximum of 6 steps.

Output signature data Cancel Continue >

IIIar 1:

Signature Generation - Step By Step

Message M to be signed:

00000	4F 6E 63 65 20 75 70 6F 6E 20 61 20 6D 69	Once upon a mi
0000E	64 6E 69 67 68 74 20 64 72 65 61 72 79 2C	dnight dreary,
0001C	20 77 68 69 6C 65 20 49 20 70 6F 6E 64 65	while I ponde
0002A	72 65 64 2C 20 77 65 61 6B 20 61 6E 64 20	red, weak and
00038	77 65 61 72 79 2C 0A 4F 76 65 72 20 6D 61	weary..Over ma
00046	6E 79 20 61 20 71 75 61 69 6E 74 20 61 6E	ny a quaint an
00054	64 20 63 75 72 69 6F 75 73 20 76 6F 6C 75	d curious volu
00062	6D 65 20 6F 66 20 66 6F 72 67 6F 74 74 65	me of forgotte

Step-by-step signature generation:

s = 7596425265155396344093287955801258962843043114986532885523799:

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

Size of message M to be signed: 6245 bytes

Continue ...

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

f = 1046107241791359592125358281218395455985348516780

Continue ...

Step 1 out of a maximum of 6 steps.

Output signature data Cancel Continue >

IIIar 2:

Signature Generation - Step By Step

Message M to be signed:

00000	4F 6E 63 65 20 75 70 6F 6E 20 61 20 6D 69	Once upon a mi
0000E	64 6E 69 67 68 74 20 64 72 65 61 72 79 2C	dnight dreary,
0001C	20 77 68 69 6C 65 20 49 20 70 6F 6E 64 65	while I ponde
0002A	72 65 64 2C 20 77 65 61 6B 20 61 6E 64 20	red, weak and
00038	77 65 61 72 79 2C 0A 4F 76 65 72 20 6D 61	weary..Over ma
00046	6E 79 20 61 20 71 75 61 69 6E 74 20 61 6E	ny a quaint an
00054	64 20 63 75 72 69 6F 75 73 20 76 6F 6C 75	d curious volu
00062	6D 65 20 6F 66 20 66 6F 72 67 6F 74 74 65	me of forgotte

Step-by-step signature generation:

f = 1046107241791359592125358281218395455985348516780

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 o

u = 6285159137953316737038225565527858208012004490808552210722300:
Vx = 123657384855345608804168786899719192392055535579248973058702:
Vy = 162443225848197255514637643368778045057855672339197438989767:

Continue ...

Step 2 out of a maximum of 6 steps.

Output signature data Cancel Continue >

Шаг 3:

Signature Generation - Step By Step

Message M to be signed:

00000	4F 6E 63 65 20 75 70 6F 6E 20 61 20 6D 69	Once upon a mi
0000E	64 6E 69 67 68 74 20 64 72 65 61 72 79 2C	dnight dreary,
0001C	20 77 68 69 6C 65 20 49 20 70 6F 6E 64 65	while I ponde
0002A	72 65 64 2C 20 77 65 61 6B 20 61 6E 64 20	red, weak and
00038	77 65 61 72 79 2C 0A 4F 76 65 72 20 6D 61	weary, .Over ma
00046	6E 79 20 61 20 71 75 61 69 6E 74 20 61 6E	ny a quaint an
00054	64 20 63 75 72 69 6F 75 73 20 76 6F 6C 75	d curious volu
00062	6D 65 20 6F 66 20 66 6F 72 67 6F 74 74 65	me of forgotte

Step-by-step signature generation:

with the domain parameters of 'EC-prime239v1' ($V=(V_x, V_y)$ is a point on

```

u = 6285159137953316737038225565527858208012004490808552210722300:
Vx = 123657384855345608804168786899719192392055535579248973058702:
Vy = 162443225848197255514637643368778045057855672339197438989767:

```

Continue ...

Convert the group element V_x (x co-ordinates of point V on elliptic curve)

```
i = 1236573848553456088041687868997191923920555355792489730587021:
```

Continue ...

Step 3 out of a maximum of 6 steps.

Output signature data Cancel Continue >

Шаг 4:

Signature Generation - Step By Step

Message M to be signed:

00000	4F 6E 63 65 20 75 70 6F 6E 20 61 20 6D 69	Once upon a mi
0000E	64 6E 69 67 68 74 20 64 72 65 61 72 79 2C	dnight dreary,
0001C	20 77 68 69 6C 65 20 49 20 70 6F 6E 64 65	while I ponde
0002A	72 65 64 2C 20 77 65 61 6B 20 61 6E 64 20	red, weak and
00038	77 65 61 72 79 2C 0A 4F 76 65 72 20 6D 61	weary, .Over ma
00046	6E 79 20 61 20 71 75 61 69 6E 74 20 61 6E	ny a quaint an
00054	64 20 63 75 72 69 6F 75 73 20 76 6F 6C 75	d curious volu
00062	6D 65 20 6F 66 20 66 6F 72 67 6F 74 74 65	me of forgotte

Step-by-step signature generation:

Continue ...

Convert the group element V_x (x co-ordinates of point V on elliptic curve)

```
i = 1236573848553456088041687868997191923920555355792489730587021:
```

Continue ...

Calculate the number $c = i \bmod r$ (c not equal to 0):

```
c = 1236573848553456088041687868997191923920555355792489730587021:
```

Continue ...

Step 4 out of a maximum of 6 steps.

Output signature data Cancel Continue >

Шаг 5:

Signature Generation - Step By Step

Message M to be signed:

00000	4F 6E 63 65 20 75 70 6F 6E 20 61 20 6D 69	Once upon a mi
0000E	64 6E 69 67 68 74 20 64 72 65 61 72 79 2C	dnight dreary,
0001C	20 77 68 69 6C 65 20 49 20 70 6F 6E 64 65	while I ponde
0002A	72 65 64 2C 20 77 65 61 6B 20 61 6E 64 20	red, weak and
00038	77 65 61 72 79 2C 0A 4F 76 65 72 20 6D 61	weary, .Over ma
00046	6E 79 20 61 20 71 75 61 69 6E 74 20 61 6E	ny a quaint an
00054	64 20 63 75 72 69 6F 75 73 20 76 6F 6C 75	d curious volu
00062	6D 65 20 6F 66 20 66 6F 72 67 6F 74 74 65	me of forgotte

Step-by-step signature generation:

Continue ...

Calculate the number $c = i \bmod r$ (c not equal to 0):

$c = 1236573848553456088041687868997191923920555355792489730587021$

Continue ...

Calculate the number $d = u^{(-1)} * (f + s * c) \bmod r$ (d not equal to 0):

$d = 8284498290377181093030551900372711591031606175177921226266381$

Continue ...

Step 5 out of a maximum of 6 steps.

Output signature data

Cancel

Continue >

Шаг 6:

Signature Generation - Step By Step

Message M to be signed:

00000	4F 6E 63 65 20 75 70 6F 6E 20 61 20 6D 69	Once upon a mi
0000E	64 6E 69 67 68 74 20 64 72 65 61 72 79 2C	dnight dreary,
0001C	20 77 68 69 6C 65 20 49 20 70 6F 6E 64 65	while I ponde
0002A	72 65 64 2C 20 77 65 61 6B 20 61 6E 64 20	red, weak and
00038	77 65 61 72 79 2C 0A 4F 76 65 72 20 6D 61	weary, .Over ma
00046	6E 79 20 61 20 71 75 61 69 6E 74 20 61 6E	ny a quaint an
00054	64 20 63 75 72 69 6F 75 73 20 76 6F 6C 75	d curious volu
00062	6D 65 20 6F 66 20 66 6F 72 67 6F 74 74 65	me of forgotte

Step-by-step signature generation:

Calculate the number $c = i \bmod r$ (c not equal to 0):

$c = 1236573848553456088041687868997191923920555355792489730587021$

Continue ...

Calculate the number $d = u^{(-1)} * (f + s * c) \bmod r$ (d not equal to 0):

$d = 8284498290377181093030551900372711591031606175177921226266381$

Continue ...

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

Step 6 out of a maximum of 6 steps.

Output signature data

Cancel

Continue >

$b = 20$
 $p = 19$
 $e1 = (11/13)$
 $d = 5$
 $e2 = d * e1 = (18/3)$
 $q = 631$

804291592781865767912555555060054978634942351405107451749311212
81330956526899

Подписание

$M = h(M) = 42$

$r = 13$

$P(u,v) = 13 * e1 = (16/18)$

$S1 = u \bmod q = 16$

$S2 = r \times h(M) + d \times S1 = 13 * 42 + 5 * 16 \bmod q = 626$

Проверка

$A = h(m)^{-1} * S1 \bmod q = 616 * 16 = 9856 \bmod q = 391$

$B = (q - s1) * m \% q = 240$

$A * e1 = 14/15$

$B * e2 = 18/16$

$+= 16/14$

$V = 16 = S1$

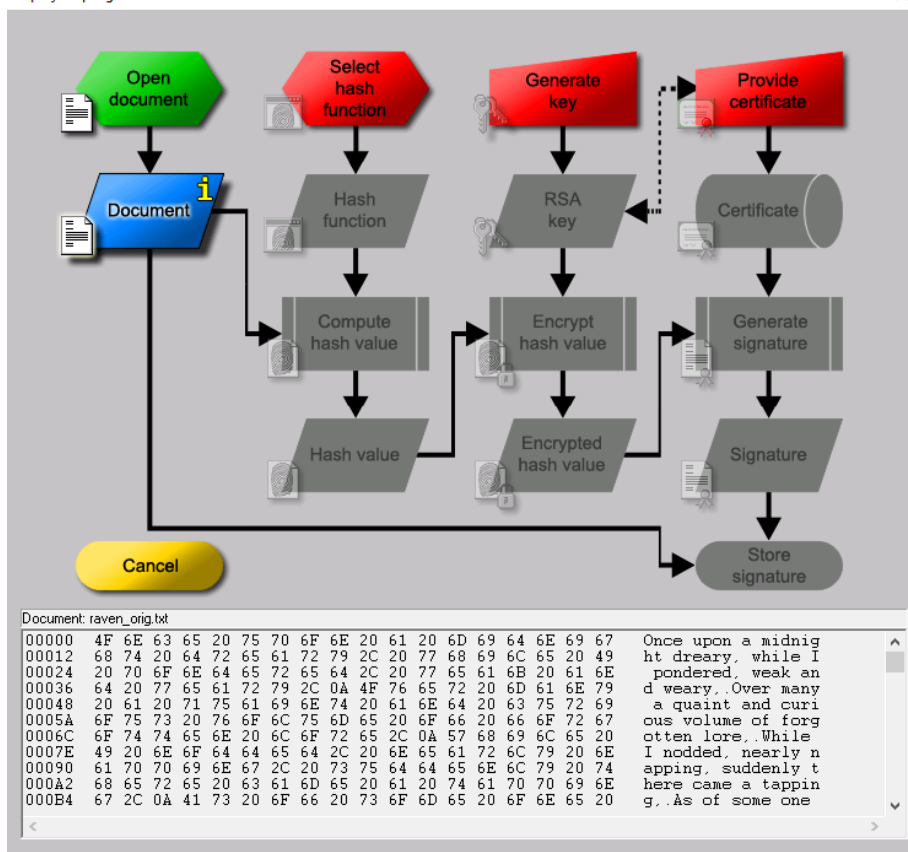
4. Демонстрация процесса подписи в среде PKI

Задание

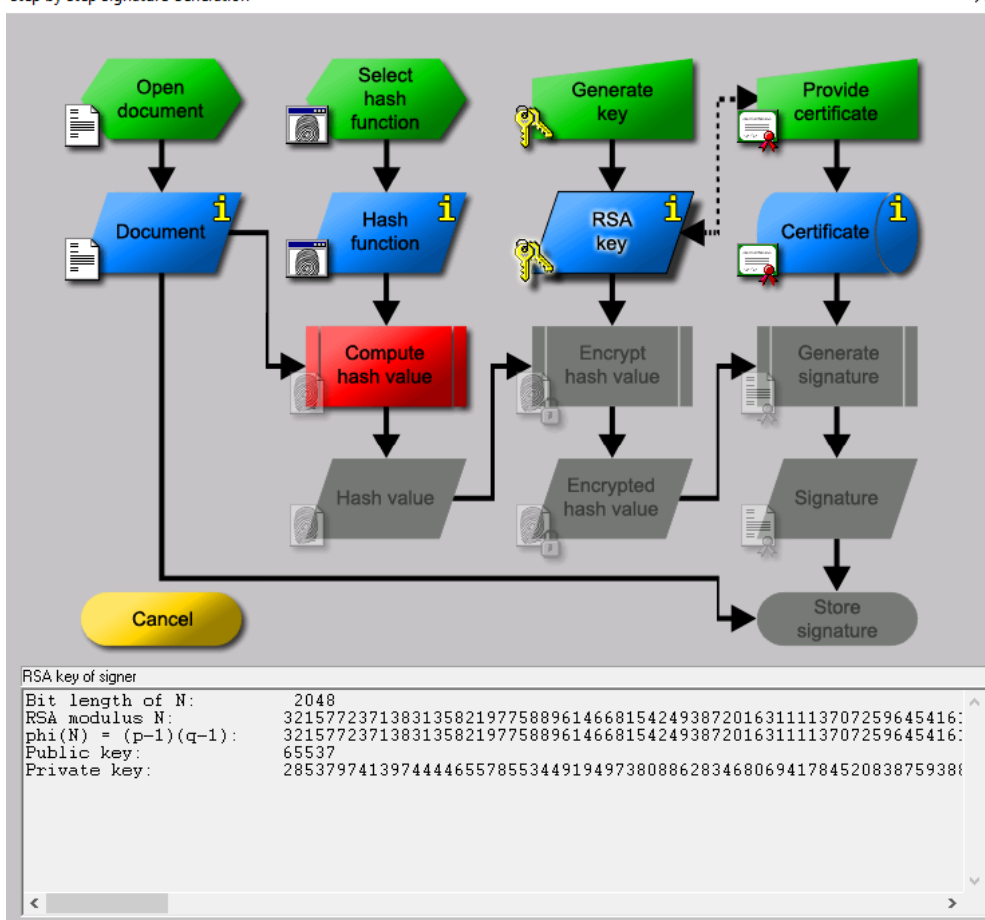
1. Запустить демонстрационную утилиту «Digital Signatures/PKI→Signature Demonstration...».
2. Получите сертификат на ранее сгенерированную ключевую пару RSA-2048.
3. Выполните и сохраните скриншоты всех этапов создания цифровой подписи документа.
4. Сохраните скриншот подписи.

Выполнение

1. Начало работы, утилита



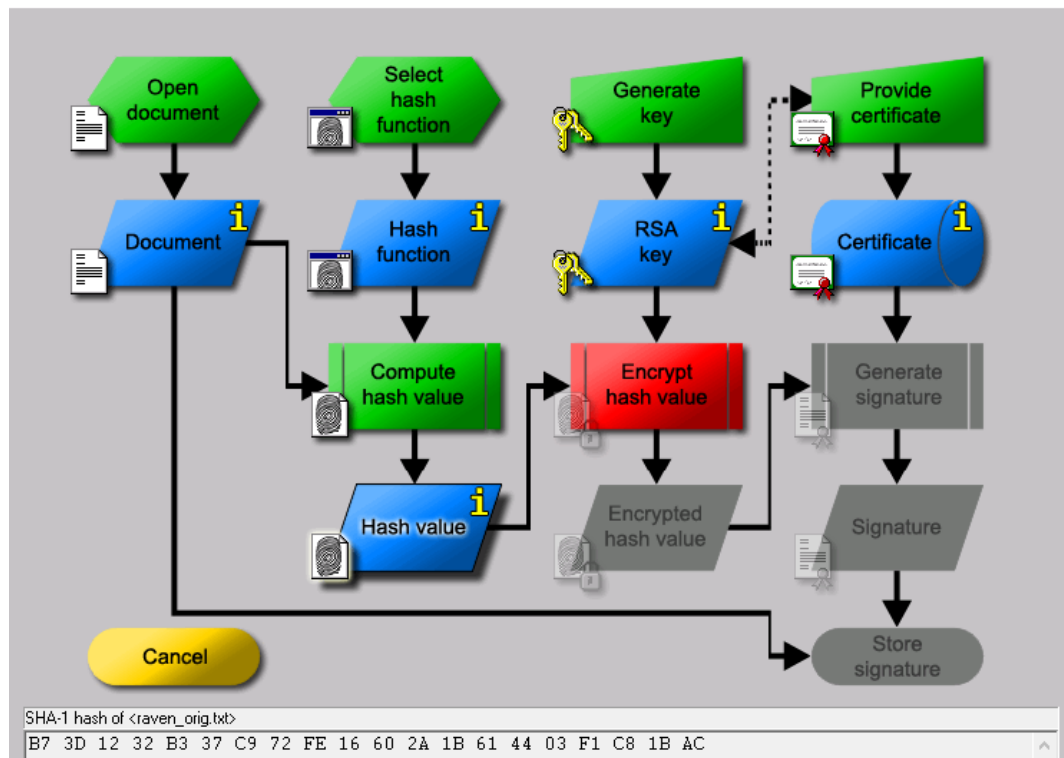
2. Использование сертификата для RSA-2048



3. Выбрана функция SHA-1

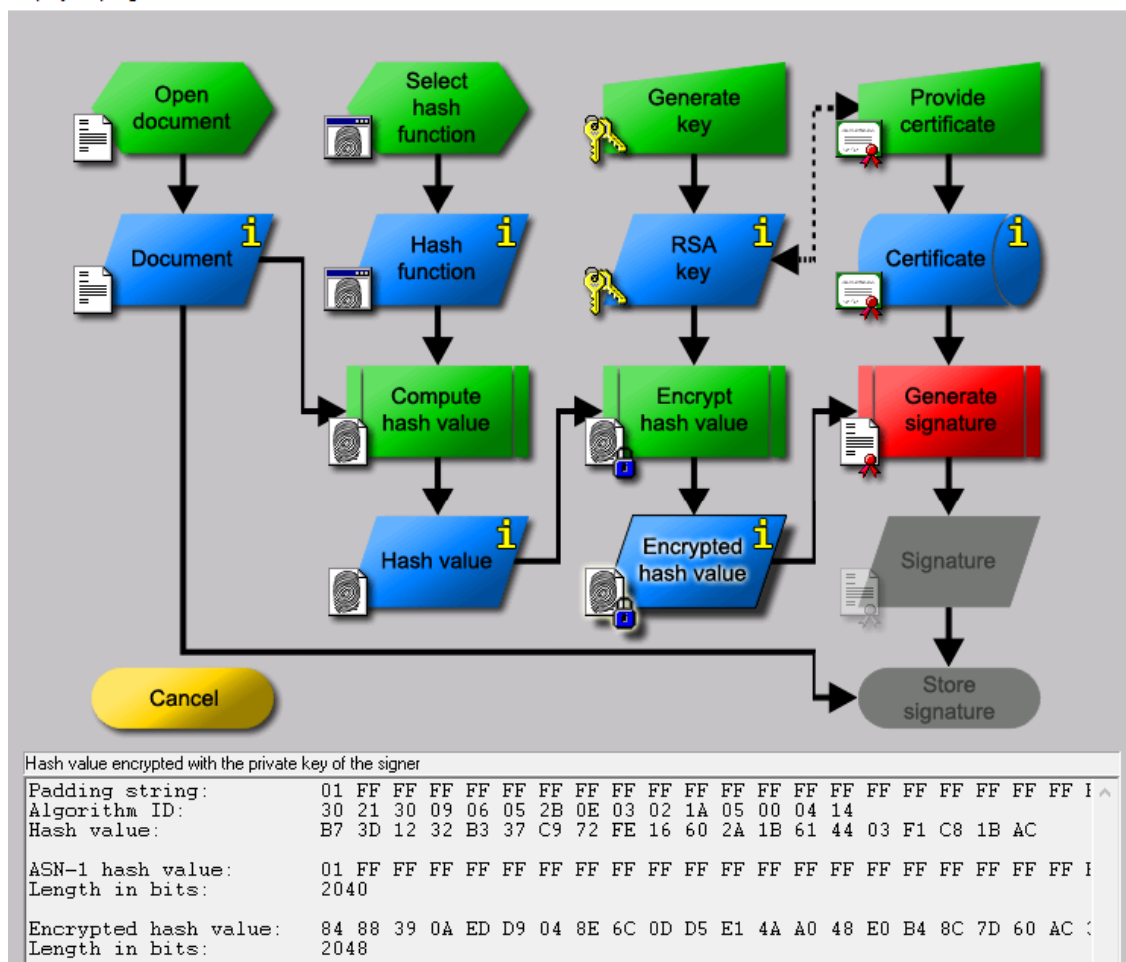
Расчитан хэш

Step by Step Signature Generation

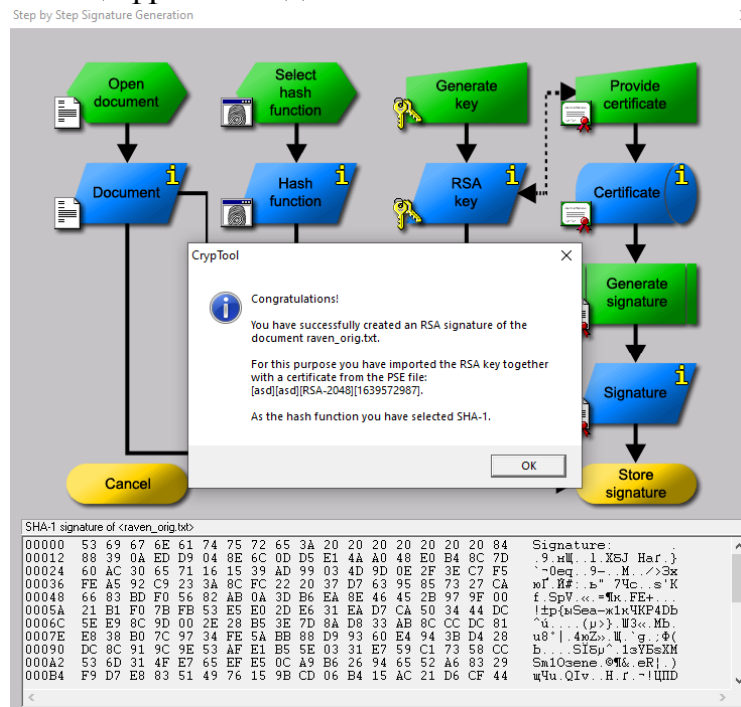


Хэш зашифрован

Step by Step Signature Generation



Сгенерирована цифровая подпись



4. Сертификат:

Create Certificate and PSE

Public RSA parameter

Bit length: 2048 bit

RSA modulus N: 321577237138313582197758896146681542493872016311113707259645

Public key e: 65537

Personal data for the certificate

Name: asd

First name: asd

Key identifier: (optional)

PIN: xxxx

PIN verification: xxxx

Generated names for PSE and certificate

User Key ID: [asd][asd][RSA-2048][1639572987]

Distinguished Name: CN=asd asd [1639572987], DC=cryptool, DC=org

Create Certificate and PSE Import certificate and key Cancel

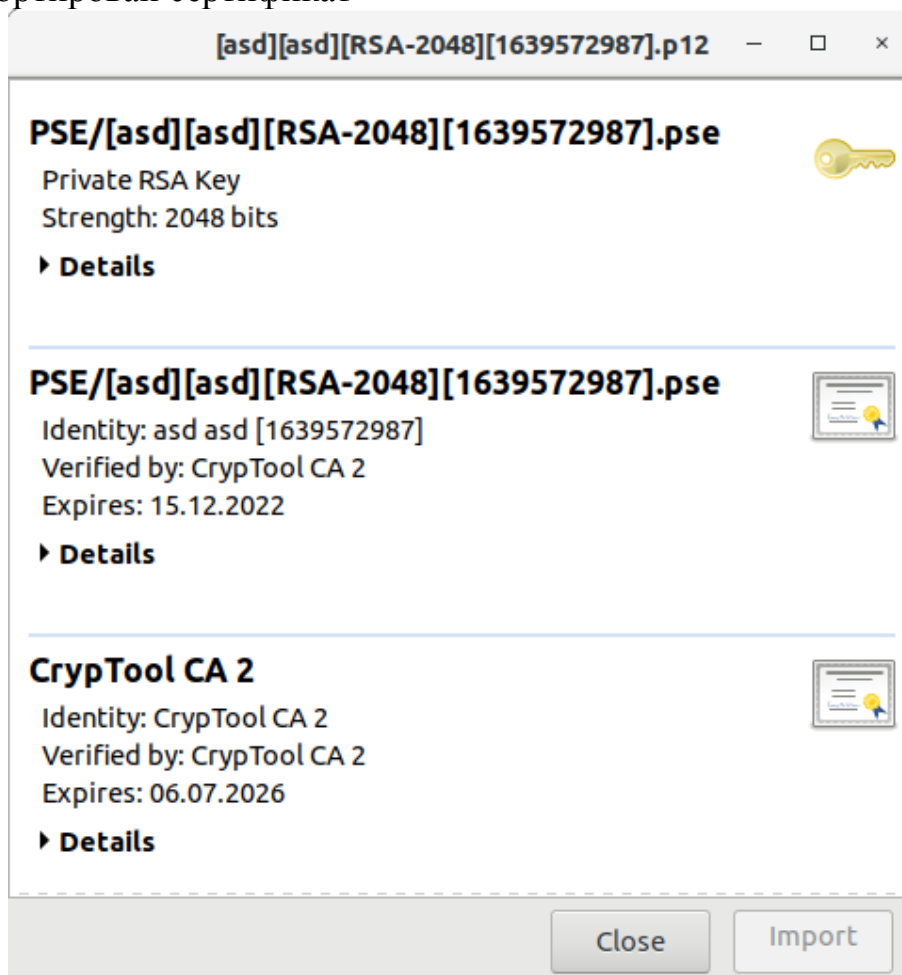
5. Имитация атаки на гибридную криптосистему

Задание

1. Сконвертируйте отчет в формат pdf.
2. Экпортируйте ранее созданный сертификат ключевой пары RSA Digital Signatures/PKI->PKI/Generate...->Export PSE(#PKCS12).
3. Откройте pdf-версию отчета и попытайтесь подписать с использованием этого сертификата.
4. Создайте собственный самоподписанный сертификат в среде Adobe Reader и используйте его для подписи отчета.
5. Сохраните скриншоты свойств подписи и сертификата.
6. Внесите изменения (маркеры, комментарии) в отчет и проверьте подпись.

Выполнение

1. На этом этапе отчет сохранён в формате ODT и экспортирован в PDF. Начиная с этой строки информация отсутствует в сохранённом отчёте
2. Экспортирован сертификат



4. Произведена подпись документа сертификатом

Sign/Encrypt Files — Kleopatra

Sign / Encrypt Files

Prove authenticity (sign)

☒ Sign as: Anton_1 (certified, OpenPGP, created: 16.12.21)



Encrypt

☐ Encrypt for others:

☐ Encrypt with password. Anyone you share the password with can read the data.

Output



☐ Encrypt / Sign each file separately.

 me/anton/Git/uni41/krypt/l8/Larin_Anton_crypt_41_lb8_5.pdf.gpg  

Sign / Encrypt **Cancel**

5. Произведена проверка корректности

Decrypt/Verify Files — Kleopatra


Output folder: /home/anton/Git/uni41/krypt/l8  

All operations completed.

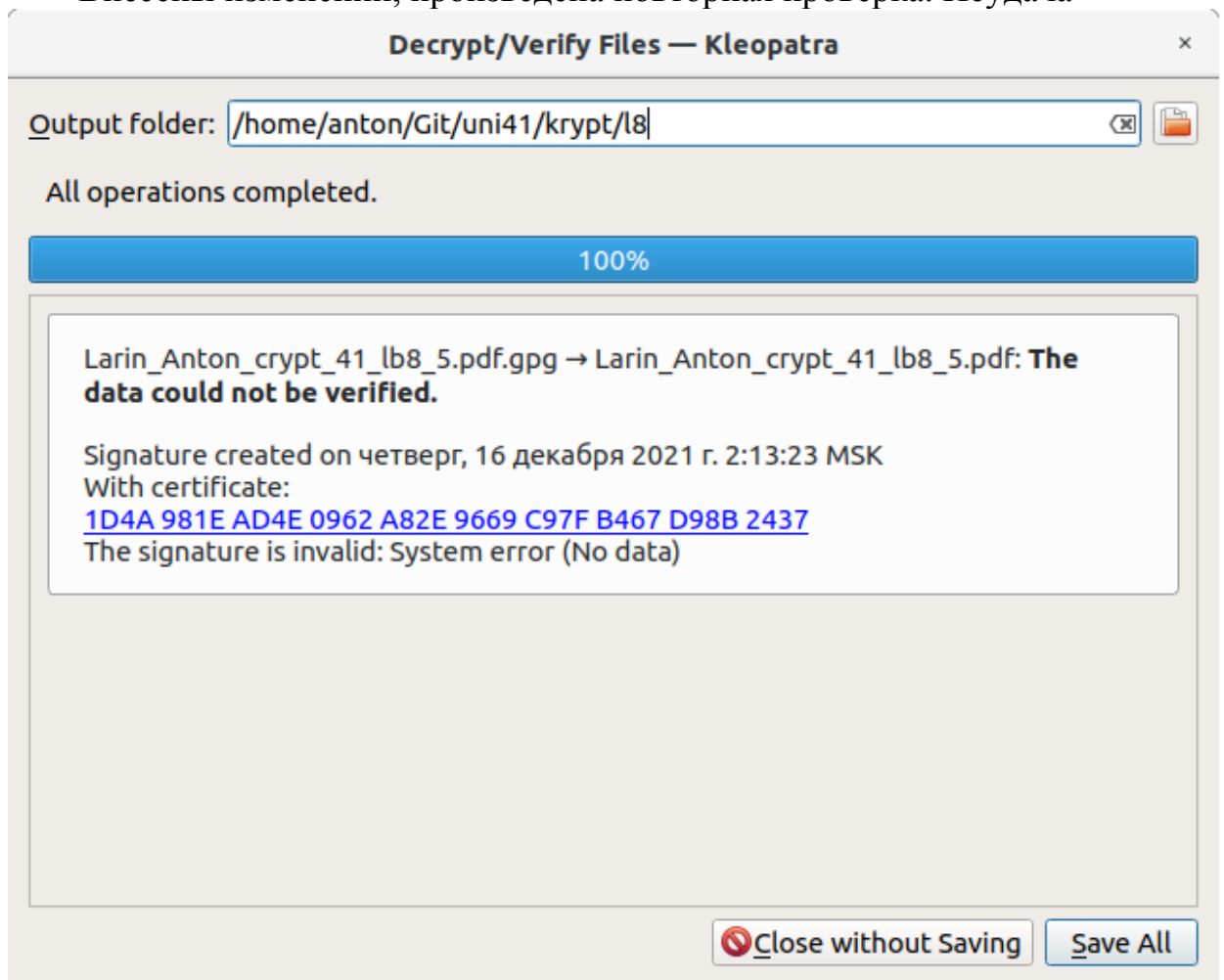
100%

Larin_Anton_crypt_41_lb8_5.pdf.gpg → Larin_Anton_crypt_41_lb8_5.pdf:
Valid signature by Anton_1

Signature created on четверг, 16 декабря 2021 г. 2:13:23 MSK
With certificate:
[1D4A 981E AD4E 0962 A82E 9669 C97F B467 D98B 2437](#)
The signature is valid and the certificate's validity is ultimately trusted.

 **Close without Saving** **Save All**

Внесены изменений, произведена повторная проверка. Неудача



Выводы.

1. Были изучены принципы работы цифровой подписи.
Сгенерированы ключевые пары по алгоритмам RSA-2048, DSA-2048, EC-239, измерено время генерации — 1, 6, <1 сек. соотв.
2. Изучен процесс создания и проверки подписи. Взят открытый текст длиной 600 символов и подписан каждой из пар ключей. Измерено время (<1 сек для всех). Сделана проверка с и без изменения текста. При изменении проверка подписи не удалась
3. Исследованы схемы на эллиптических кривых.
Выполнено создание подписи алгоритмом ECSP-DSA в пошаговом режиме при помощи Cryptool. Для этого использована пара ключей из первого задания. Выполнена проверка целостности с /без изменений.
Выполнена ручная проверка алгоритма ECDSA при помощи инструмента работы с точками на эллиптической кривой в Cryptool. Последовательно выполнена генерация ключей, подпись, и проверка подписи, согласно лекционному материалу. Результат проверки показал, что подпись была верна
4. Проведена симуляция подписи в среде PKI, (с использованием инфраструктуры ключей). На сгенерированную пару ключей получен сертификат
5. Выполнено подписание данного (не законченного) отчёта при помощи сгенерированного ранее сертификата. Сделана попытка фальсификации(изменения) отчёта. Это привело в неудачной попытке валидации отчёта при помощи сертификата