

Отчет по лабораторной работе №1
Система верстки TEX и расширения LATEX

Греченко Лаура

1 июня 2015 г.

1. Создание минимального файла .tex в простом текстовом редакторе – преамбула, тело документа.

Документ LaTeX — это текстовый файл, содержащий специальные команды языка разметки. Сам документ делится на преамбулу и тело. Преамбула содержит информацию про класс документа, использованные пакеты макросов, определения макросов, автора, дату создания документа и другую информацию. Тело документа содержит собственно текст документа и команды разметки. Оно должно находиться между командами

```
\begin{document} , \end{document}
```

Создали файл lab1.tex, сохранили в папке.

2. Компиляция в командной строке – latex, xdvi, pdflatex.

Компилируем .tex сразу в .pdf.

```
pdflatex -aux-directory=E:/Study/Z/Result -output-directory=  
E:/Study/ZI/Result E:/Study/ZI/Result/lab1.tex
```

Компилируем .tex в файл .dvi.

```
latex -aux-directory=E:/Study/ZI/Result -output-directory=  
E:/Study/ZI/Result E:/Study/ZI/Result/lab1.tex
```

Затем из .dvi получили .pdf.

```
xdvipdfmx -o E:/Study/ZI/Result/lab1.pdf E:/Study/ZI/Result/lab1.dvi
```

3. Оболочка TexMaker, Быстрый старт, Быстрая сборка.

Texmaker - это одна из нескольких популярных "оболочек или "сред разработок т.е. программ, объединяющих в себе текстовый редактор и простой графический интерфейс к программам системы TeX, таким как latex или pdflatex. Чтобы задать преамбулу документа, можем использовать помощника "Быстрый старт"(Меню "Помощник"). Самый простой способ скомпилировать документ это использовать команду "Быстрая сборка". Можем задать последовательность команд используемую командой "Быстрая сборка" в диалоге "Настроить Texmaker". Наиболее популярной реализацией системы TeX и ее многочисленных расширений для ОС Windows является MiKTeX. В данной лабораторной работе использовался MikTeX.

4. Создание титульного листа, нескольких разделов, списка, несложной формулы.

Класс article включает следующие команды секционирования:

```
\section{...}  
\subsection{...}  
\subsubsection{...}  
\paragraph{...}  
\subparagraph{...}
```

Нумерованные списки:

```
begin{enumerate}  
\item  
\end{enumerate}
```

Титульный лист:

`\title{...}`

Пример несложных формул:

$$e = m \cdot c^2; \quad (1)$$

$$\sum_{k=1}^n I_k = 0 \quad (2)$$

5. Понятие классов документов, подключаемых пакетов. При обработке входного файла, Latex должен знать тип создаваемого документа. Он задается командой

`\documentclass[опции]{класс}`

Здесь класс определяет тип создаваемого документа. В состав Latex входят дополнительные классы для других документов, включая письма и слайды. Параметр опции изменяет поведение класса документа. Для включения в документ графики, цветного текста или исходного кода программы из внешнего файла, необходимо расширить возможности Latex. Такие расширения называются пакетами. Пакеты активизируются командой

`\usepackage[опции]{класс}`

Где пакет - это имя пакета, а опции - список ключевых слов, включающих специальные свойства пакета.

6. Верстка более сложных формул.
Примеры сложных формул:

$$a^+y \neq a^{x+y}; \quad (3)$$

$$v = \sigma_1 \cdot \sigma_2 \tau_1 \cdot \tau_2;$$
$$\sum_{i=1}^n \int_0^{\frac{\pi}{2}} \prod_{\epsilon}$$

Выводы:

1. + Большое количество пакетов и макетов.
2. - Создание нового макета занимает много времени.

При работе с LATEX потребовалось много времени для ознакомления с основными командами.

Отчет по лабораторной работе №2
Система контроля версий Git

1. Изучить справку для основных команд.
В ходе данной лабораторной работы были изучены материалы из списка рекомендованных.
2. Получить содержимое репозитория.
Создали на github тестовый репозиторий.
`git clone git:/github.com/luaraAmsterdam/TestRepositories`
3. Добавить новую папку и первого файла под контроль версий
С помощью команд `mkdir` и `cat` создали папку `src` и файл `test.txt`.

```
Laura@LA /e/Study/ZI/Result/2/GitHub/TestRepositories (master)
$ ls -l
total 1
-rw-r--r--  1 Laura  Administ  20 May 25 02:06 README.md
drwxr-xr-x  2 Laura  Administ   0 May 25 02:15 src
-rw-r--r--  1 Laura  Administ  21 May 25 02:37 test.txt
```

4. Зафиксировать изменения в локальном репозитории.
Добавляем файлы рекурсивно в директории.
`git add`

```
Laura@LA /e/Study/ZI/Result/2/GitHub/TestRepositories (master)
$ git add .

Laura@LA /e/Study/ZI/Result/2/GitHub/TestRepositories (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   test.txt
```

5. Внести изменения в файл и просмотреть различия.
Показывает различия между сохраненными данными и не сохраненными изменениями.
`git diff HEAD test.txt`

```
Laura@LA /e/Study/ZI/Result/2/GitHub/TestRepositories (master)
$ git diff HEAD test.txt
diff --git a/test.txt b/test.txt
new file mode 100644
index 0000000..51033e3
--- /dev/null
+++ b/test.txt
@@ -0,0 +1,5 @@
+created new file
+.
+edited file
+delete edit
+edit again
\ No newline at end of file
```

6. Отменить локальные изменения.
Возвращаем файл в исходное состояние.
`git reset HEAD test.txt`
7. Внести изменения в файл и просмотреть различия.
Просматриваем различия между сохраненными данными и не сохраненными изменениями.
`git diff HEAD test.txt`
8. Зафиксировать изменения в локальном репозитории, зафиксировать изменения в центральном репозитории
`git add .`
`git commit -m "test commit"`
`git push origin master`

```
Laura@LA /e/Study/ZI/Result/2/GitHub/TestRepositories (master)
$ git push origin master
Enter passphrase for key '/c/Users/Laura/.ssh/id_rsa':
Counting objects: 9, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (7/7), done.
Writing objects: 100% (8/8), 750 bytes | 0 bytes/s, done.
Total 8 (delta 1), reused 0 (delta 0)
To git@github.com:luaraAmsterdam/TestRepositories
9f31a8a..f1200f8 master -> master
```

9. Получить изменения из центрального репозитория
`git pull`

```
Laura@LA /e/Study/ZI/Result/2/GitHub/TestRepositories (master)
$ git pull
Enter passphrase for key '/c/Users/Laura/.ssh/id_rsa':
Updating f1200f8..6b84eb2
Fast-forward
 create_in_github.txt | 2 ++
 1 file changed, 2 insertions(+)
 create mode 100644 create_in_github.txt
```

10. Поэкспериментировать с ветками
Создаем новую ветку.
`git branch testt`
Переключаем рабочую версию на указанную ветку.
`git checkout testt`
Создаем изменения в ветке.
`vim testtt.swp`
`git add .`
`git commit -m "testt commit"`
Переключаем на ветку master.
`git checkout master`
Применяем изменения из указанной ветки.
`git merge testt`

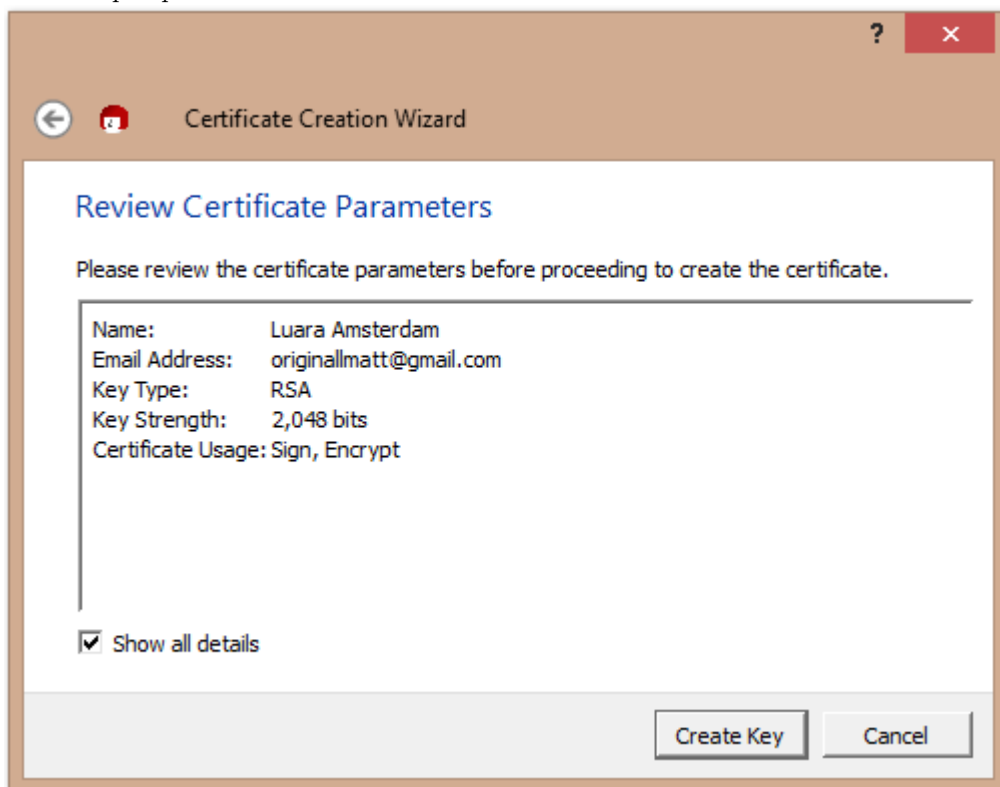
```
Laura@LA /e/Study/ZI/Result/2/GitHub/TestRepositories (master)
$ git merge testt
Updating 6b84eb2..a87aedb
Fast-forward
 .test3.txt.swp | Bin 0 -> 12288 bytes
 .testtt.swp    | Bin 0 -> 12288 bytes
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 .test3.txt.swp
 create mode 100644 .testtt.swp
```

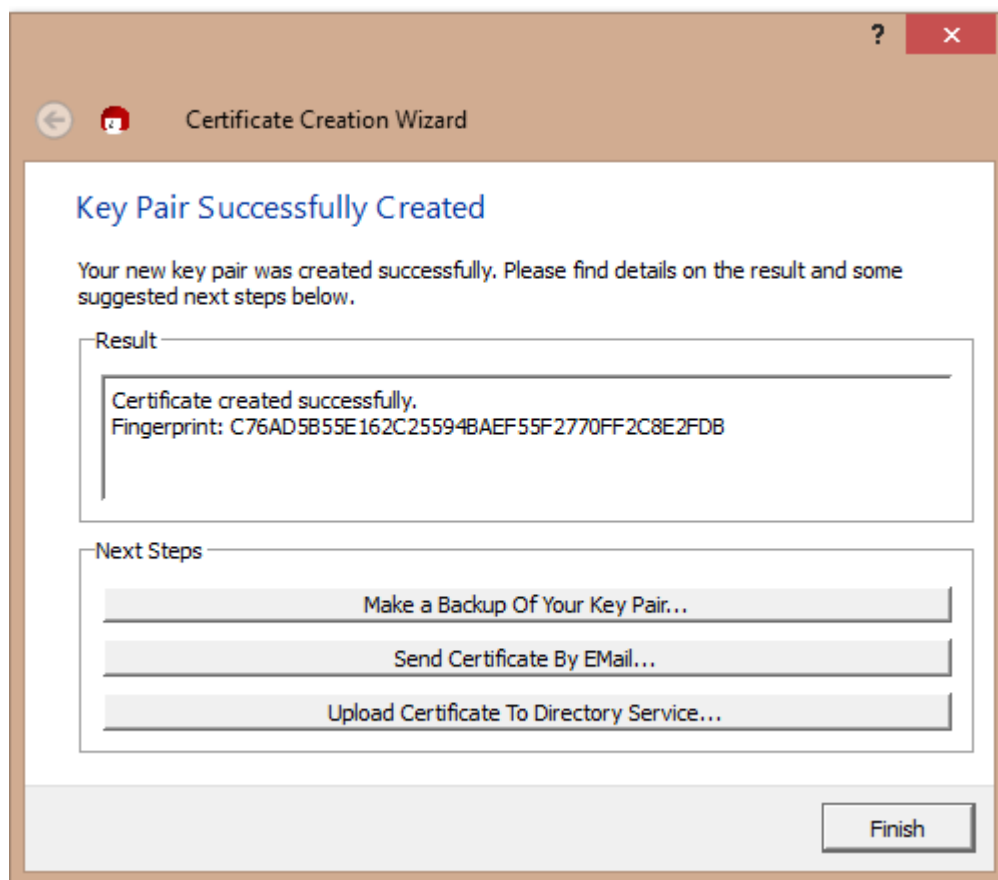
Вывод:

В ходе данной лабораторной работы мы изучили работу с git. Полученные навыки полезны и актуальны. Система применяется для сохранений изменений файлов с возможностью восстановления старых версий.



Отчет по лабораторной работе №3
Программа для шифрования и подписи GPG, пакет Gpg4win

1. Изучить документацию, запустить графическую оболочку Kleopatra
Была изучена документация перечисленная в разделе материалы.
2. Создать ключевую пару OpenPGP (File => New Certificate) Создали новый сертификат.





3. Экспортировать сертификат (File => Export Certificate) Экспортировали сертификат.
4. Поставить ЭЦП на файл (File => Sign/Encrypt Files)
Выбираем пункт подписать и зашифровать.

 Sign/Encrypt Files

What do you want to do?

Please select here whether you want to sign or encrypt files.

Selected file:

- E:\Study\ZI\Result\3\sign_file.txt

☐ Archive files with: TAR (PGP®-compatible)

Archive name (OpenPGP): E:\Study\ZI\Result\3\sign_file.txt.tar

Archive name (S/MIME): E:\Study\ZI\Result\3\sign_file.txt.tar.gz

☒ Sign and Encrypt (OpenPGP only)

☐ Encrypt

☐ Sign

☐ Text output (ASCII armor)

☐ Remove unencrypted original file when done

Next

Cancel

Выбираем сертификат.

Sign/Encrypt Files

For whom do you want to encrypt?

Please select for whom you want the files to be encrypted. Do not forget to pick one of your own certificates.

Search...

All Certificates

Name	E-Mail	Valid From	Valid Until
<input type="checkbox"/> Luara Amsterdam	originalmatt@gmail.com	2015-05-25	
<input type="checkbox"/> Karina Vilegzhanina	k.vilegzhanina@gmail.com	2015-02-08	

▼ Add

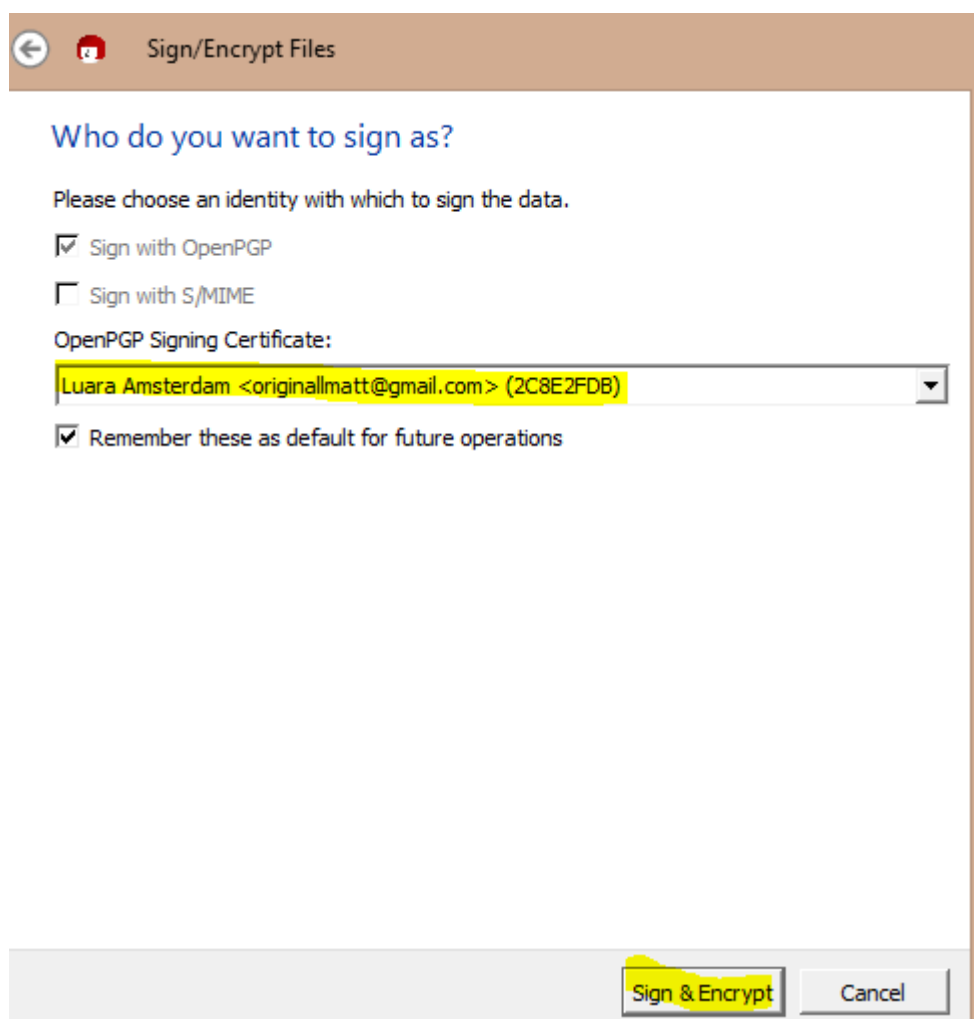
▲ Remove

Name	E-Mail	Valid From	Valid Until
<input type="checkbox"/> Luara Amsterdam	originalmatt@gmail.com	2015-05-25	

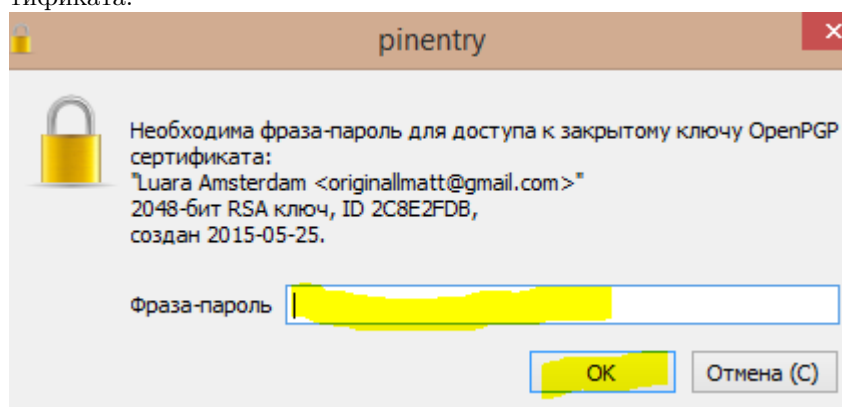
Next

Cancel

Выбираем сертификат OpenPGP.



Необходимо ввести фразу-пароль, введенную при формировании сертификата.



В результате успешно зашифровали и подписали данные.

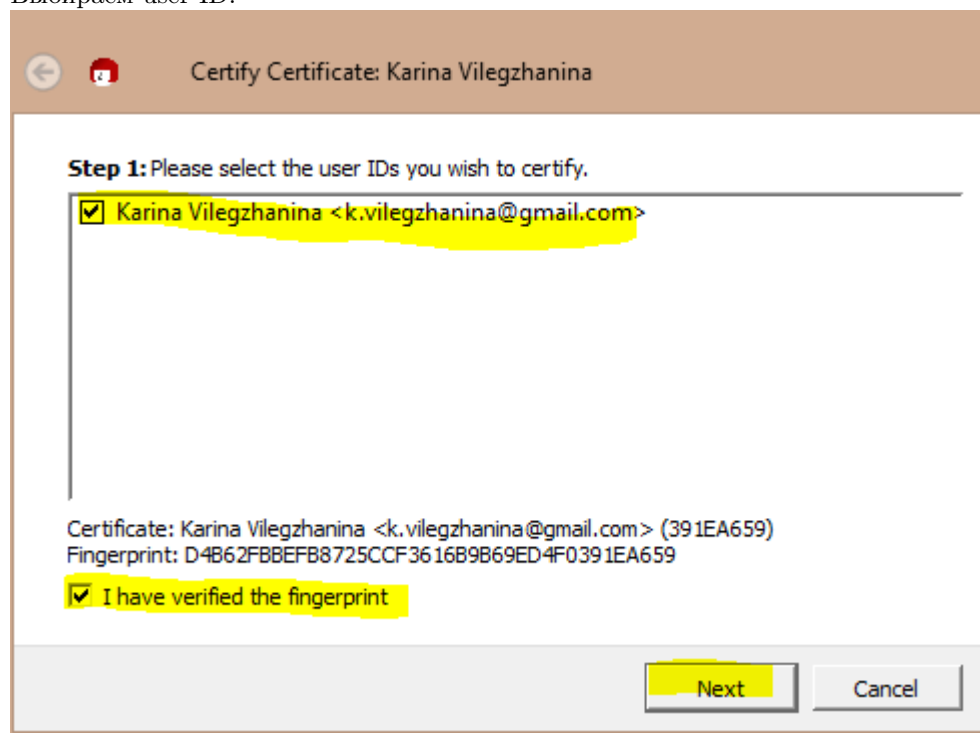
Results

Status and progress of the crypto operations is shown here.

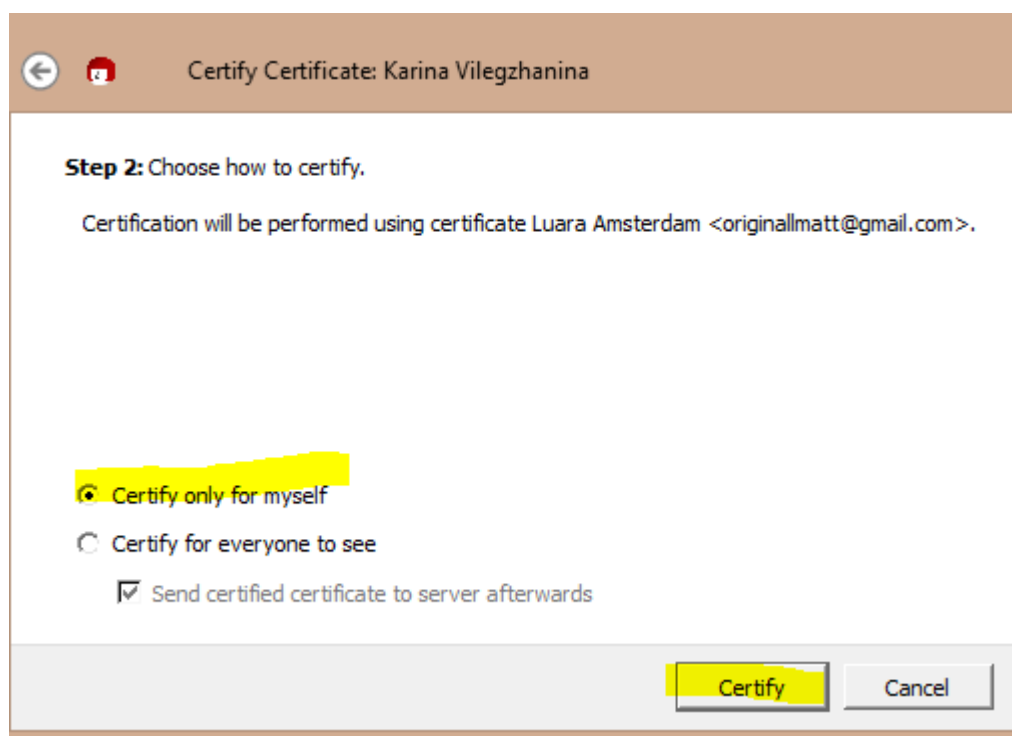
OpenPGP: All operations completed.

sign_file.txt → sign_file.txt.gpg: **Signing and encryption succeeded.**

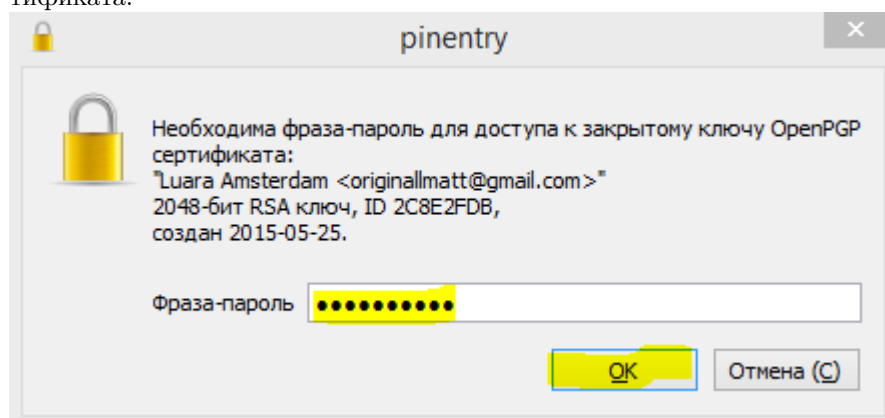
5. Получить чужой сертификат из репозитория
Скачали сертификат с <https://github.com/vilegzhanina/InfoSecCourse201>,
файл с данными и файл с сигнатурой (подписью).
6. Импортировать сертификат, подписать его
Выбираем user ID.



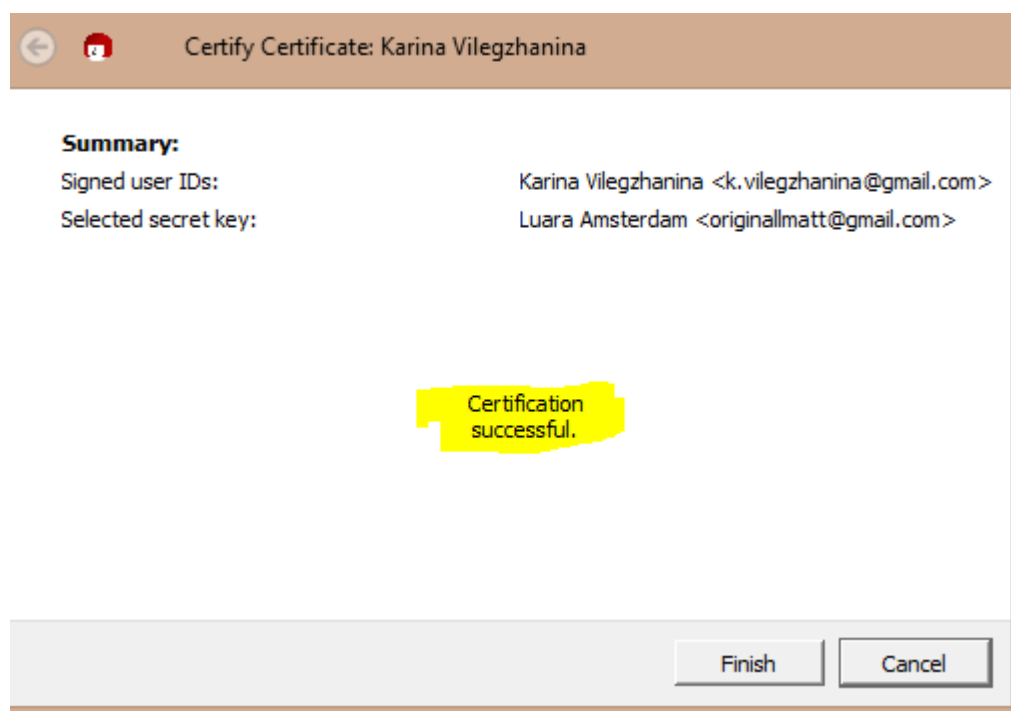
Сертифицируем только для себя.



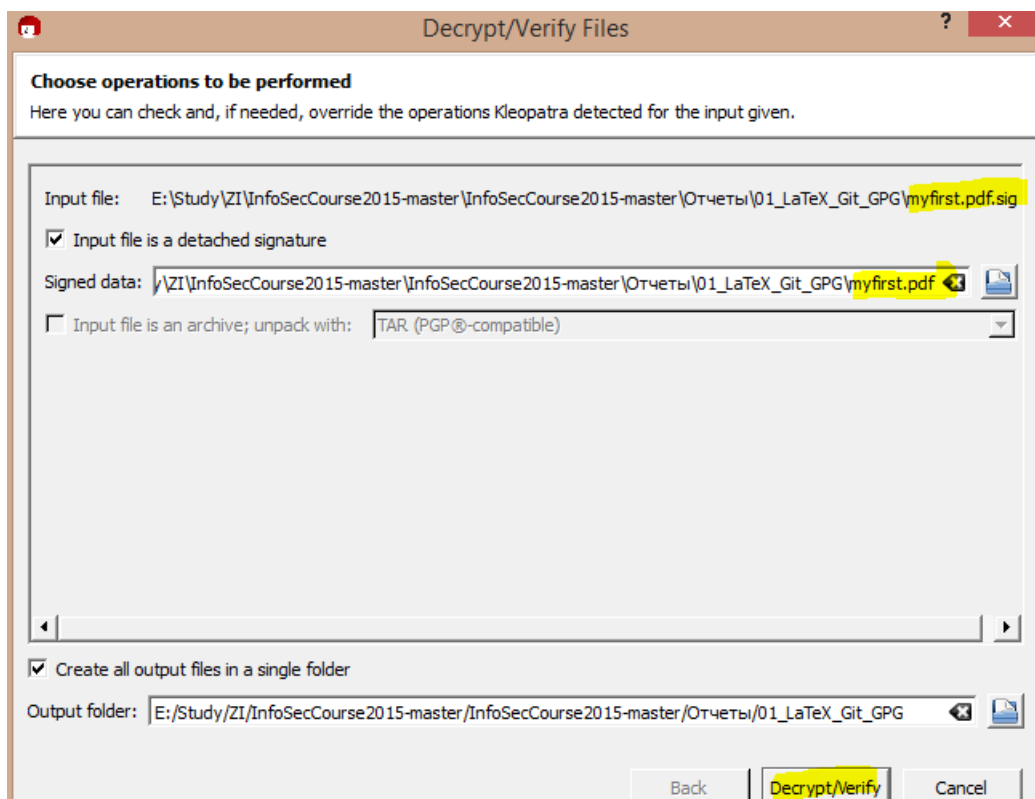
Необходимо ввести фразу-пароль, введенную при формировании сертификата.

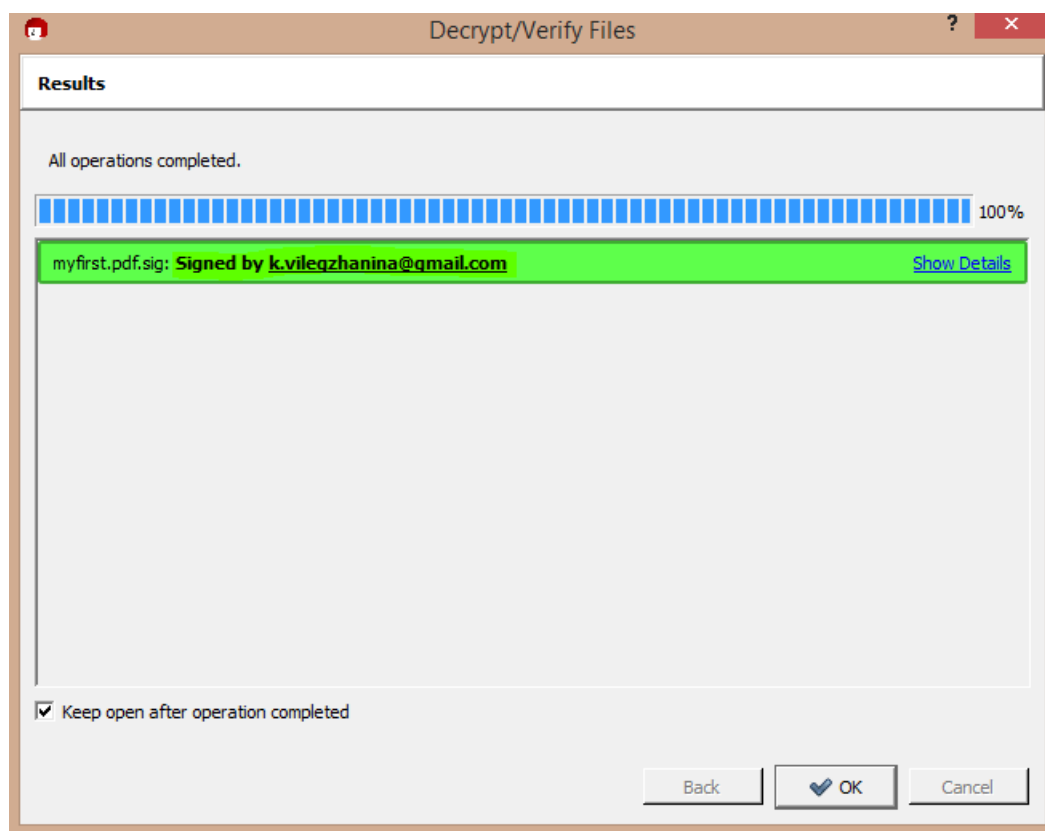


В результате успешно выполнили сертификацию.



7. Проверить подпись
Верифицируем скаченный файл с данными и файл с подписью.





8. Взять сертификат кого-либо из коллег, зашифровать и подписать для него какой-либо текст, предоставить свой сертификат, убедиться, что ему удалось получить открытый текст, проверить подпись Подписали и зашифровали данные для коллеги (Певцов И.).

Sign/Encrypt Files

For whom do you want to encrypt?

Please select for whom you want the files to be encrypted. Do not forget to pick one of your own certificates.

Name	E-Mail
Windows Remote Shaman (www.remoteshaman.com)	remoteshaman.com@gr
Pevtsov	huxley92@mail.ru
Luara Amsterdam (test)	originallmatt@gmail.co
Luara Amsterdam	originallmatt@gmail.co
Karina Vilegzhanina	k.vilegzhanina@gmail.co

Name	E-Mail	Valid From	Valid Until	Details
Pevtsov	huxley92@mail.ru	2015-05-25		OpenPGP 24

Results

Status and progress of the crypto operations is shown here.

OpenPGP: All operations completed.

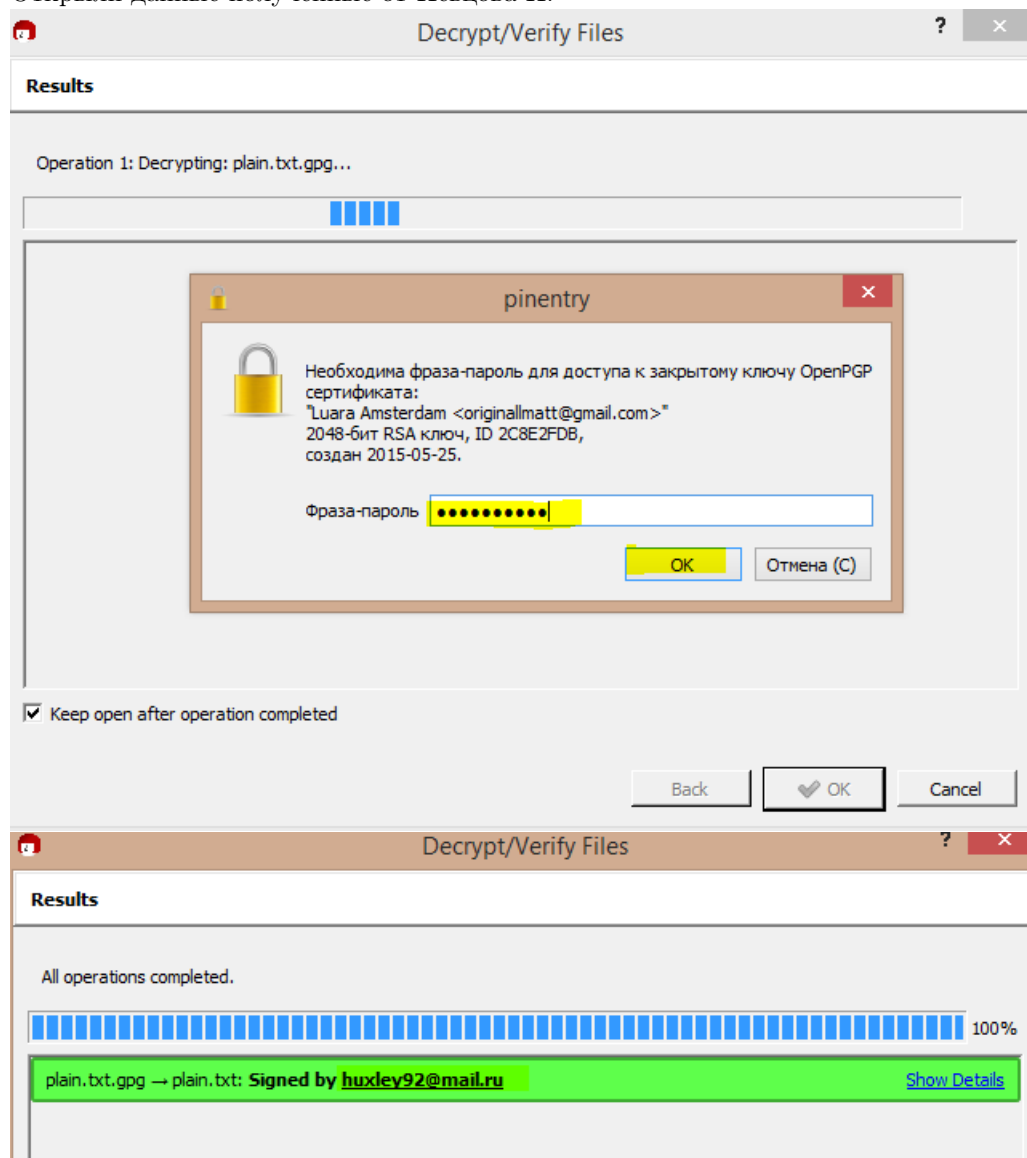
sign_file.txt → sign_file.txt.gpg: **Signing and encryption succeeded.**

☒ Keep open after operation completed

Певцов И. подтвердил успешное открытие данных.

9. Предыдущий пункт наоборот.

Открыли данные полученные от Певцова И.



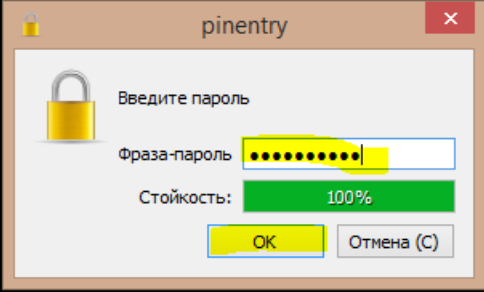
10. Используя GNU Privacy handbook (ссылка в материалах) потренироваться в использовании gpg через интерфейс командной строки, без использования графических оболочек.

Создание пары PGP ключей

```
E:\Study\ZI\Result\3\cmd>gpg --gen-key
gpg (GnuPG) 2.0.27; Copyright (C) 2015 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Выберите тип ключа:
(1) RSA и RSA (по умолчанию)
(2) DSA и Elgamal
(3) DSA (только для подписи)
(4) RSA (только для подписи)
Ваш выбор? 1
Ключи RSA могут иметь длину от 1024 до 4096 бит.
Какой размер ключа Вам необходим? (2048)
Запрошенный размер ключа - 2048 бит
Выберите срок действия ключа.
  0 = без ограничения срока действия
  <n> = срок действия - n дней
  <n>w = срок действия - n недель
  <n>m = срок действия - n месяцев
  <n>y = срок действия - n лет
Срок действия ключа? (0) 6m
Ключ действителен до: 11/21/15 17:02:06 Russia TZ 2 Standard Time
Все верно? (y/N) y

GnuPG необходимо составить ID пользователя в качестве идентификатора ключа.
Ваше настоящее имя: Luara Amsterdam
Адрес электронной почты: originallmatt@gmail.com
Комментарий: test
Вы выбрали следующий ID пользователя:
  "Luara Amsterdam (test) (originallmatt@gmail.com)"
Сменить (N)Имя, (C)Комментарий, (E)Адрес или (O)Принять/(Q)Выход? O
Для защиты закрытого ключа необходима фраза-пароль.
```



Получение публичного PGP ключа

Шифровка и обмен файлами с использованием публичных ключей

```

E:\Study\ZI\Result\3\cmd>gpg --keyserver keys.gnupg.net --recv-keys 346B72D7
gpg: запрашиваю ключ 346B72D7 с hkp сервера keys.gnupg.net
gpg: DBG: armor-keys-failed <KEY 0x346B72D7 BEGIN
> ->0
gpg: DBG: armor-keys-failed <KEY 0x346B72D7 END
> ->0
gpg: ключ 346B72D7: импортирован открытый ключ "Windows Remote Shaman <www.remoteshaman.com> <GPG key for remoteshaman.com@gmail.com email> <remoteshaman.com@gmail.com>"
gpg: Всего обработано: 1
gpg: импортировано: 1 (RSA: 1)

E:\Study\ZI\Result\3\cmd>gpg -e -r 346B72D7 sign_file.txt
gpg: 6D2314B5: Нет свидетельств того, что данный ключ принадлежит названному пользователю

pub 4096R/6D2314B5 2014-01-19 Windows Remote Shaman <www.remoteshaman.com> <GPG key for remoteshaman.com@gmail.com email> <remoteshaman.com@gmail.com>
Отпечаток главного ключа: 8CC0 592D 17F5 0B9D A625 3324 1590 B040 346B 72D7
Отпечаток подключа: F711 E6D7 034B 4D58 B324 DFB2 6B94 B164 6D23 14B5

Нет уверенности в том, что ключ принадлежит человеку, указанному в ID пользователя ключа. Если Вы ТОЧНО знаете, что делаете, можете ответить на следующий вопрос утвердительно.

Все равно использовать данный ключ? (y/N) y

```

Вывод списка ключей.

```

E:\Study\ZI\Result\3\cmd>gpg --list-keys
C:/Users/Laura/AppData/Roaming/gnupg/pubring.gpg
-----
pub 2048R/2C8E2FDB 2015-05-25
uid [абсолютное] Luara Amsterdam <originallmatt@gmail.com>

pub 2048R/391EA659 2015-02-08
uid [ полное ] Karina Vilegzhanina <k.vilegzhanina@gmail.com>

pub 2048R/9CEC5726 2015-05-25 [срок действия истекает: 2015-11-21]
uid [абсолютное] Luara Amsterdam (test) <originallmatt@gmail.com>
sub 2048R/3EDF1F75 2015-05-25 [срок действия истекает: 2015-11-21]

pub 4096R/346B72D7 2014-01-19
uid [неизвестно] Windows Remote Shaman <www.remoteshaman.com> <GPG key for remoteshaman.com@gmail.com email> <remoteshaman.com@gmail.com>
sub 4096R/6D2314B5 2014-01-19

```

Для расшифрования полученных данных необходимо знать приватный ключ.

```

E:\Study\ZI\Result\3\cmd>gpg -d sign_file.txt.gpg > new_file.txt
gpg: зашифровано 4096-битным ключом RSA, с ID 6D2314B5, созданным 2014-01-19
"Windows Remote Shaman <www.remoteshaman.com> <GPG key for remoteshaman.com@gmail.com email> <remoteshaman.com@gmail.com>"
gpg: свой расшифровки: No secret key

```

Экспорт/импорт PGP (GnuPG) ключей

Приведённая выше команда выполнит экспорт публичного PGP ключа (с ИД 346B72D7) в файл 346B72D7.public.gpg в двоичном (binary) формате, но это может быть неудобно при его пересылке например в теле сообщения по электронной почте. Мы можем выполнить экспорт ключа в ASCII формате добавив флаг `-armor` (или просто `-a`).

```

E:\Study\ZI\Result\3\cmd>gpg --export --armor --output 9CEC5726.public.gpg 9CEC5726
E:\Study\ZI\Result\3\cmd>gpg --export-secret-keys -a --output 9CEC5726.private.gpg 9CEC5726

```

Вывод:

В ходе данной лабораторной работы мы научились создавать сертификаты, шифровать файлы и ставить ЭЦП. Использовались инструменты графическая оболочка Kleopatra, утилита gpg.